

ГУАП

КАФЕДРА № 42

ОТЧЕТ  
ЗАЩИЩЕН С ОЦЕНКОЙ \_\_\_\_\_

ПРЕПОДАВАТЕЛЬ

Доцент				Бржезовский А. В.
должность, уч. степень, звание		подпись, дата		инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №1

СОЗДАНИЕ ТАБЛИЦ БАЗЫ ДАННЫХ

Вариант 5

по курсу: Методы и средства проектирования информационных систем  
и технологий

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №	4128			Воробьев В.А.
			подпись, дата	инициалы, фамилия

## СОДЕРЖАНИЕ

<b>1</b>	<b>Постановка задачи . . . . .</b>	<b>3</b>
<b>2</b>	<b>Выполнение работы . . . . .</b>	<b>4</b>
<b>3</b>	<b>Вывод . . . . .</b>	<b>7</b>
	<b>Приложение . . . . .</b>	<b>8</b>

## **1 Постановка задачи**

### **Цель работы:**

Приобретение навыков создания бд и таблиц, а также настройка правил и ссылочной целостности.

### **Задание:**

Произвести создание БД и таблиц, определив правила проверки значений и задав ограничения ссылочной целостности. Структура БД должна обеспечивать хранение сведений, необходимых для выполнения запросов, указанных в варианте задания.

### **Содержание отчета:**

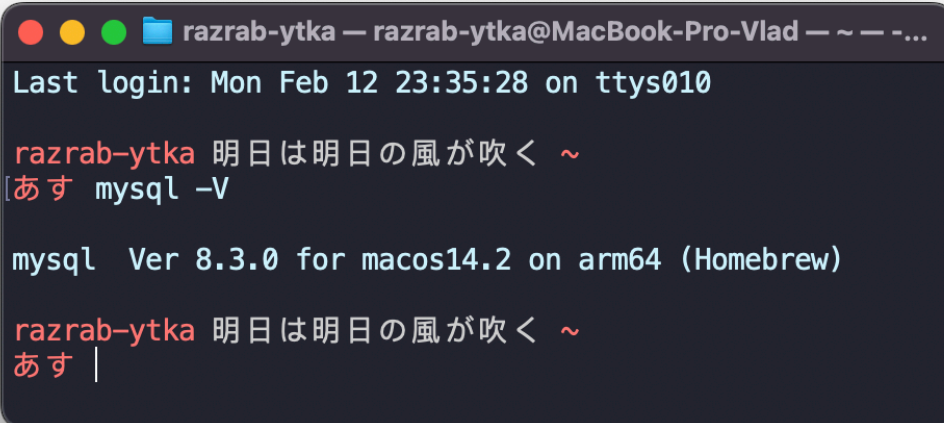
1. схема БД;
2. скрипт SQL для создания таблиц;
3. описание структуры таблиц, ограничений на значения данных, ссылочной целостности, реализованных в БД.

### **Вариант задания:**

Создайте базу данных для хранения следующих сведений: ВУЗ, студент, группа, факультет, конференция, тема доклада, программа конференции.

## 2 Выполнение работы

Для выполнения работы была выбрана СУБД MySQL. Версия программного обеспечения представлена на рисунке 1.

A screenshot of a macOS terminal window. The title bar shows the window name 'razrab-ytka' and the user 'razrab-ytka@MacBook-Pro-Vlad'. The terminal content shows a login message 'Last login: Mon Feb 12 23:35:28 on ttys010', followed by a prompt 'razrab-ytka' and a command 'mysql -V'. The output of the command is 'mysql Ver 8.3.0 for macos14.2 on arm64 (Homebrew)'. The prompt 'razrab-ytka' appears again at the bottom.

```
razrab-ytka — razrab-ytka@MacBook-Pro-Vlad — ~ — ...  
Last login: Mon Feb 12 23:35:28 on ttys010  
razrab-ytka 明日は明日の風が吹く ~  
[あす mysql -V  
mysql Ver 8.3.0 for macos14.2 on arm64 (Homebrew)  
razrab-ytka 明日は明日の風が吹く ~  
あす |
```

Рисунок 1 - Версия программного обеспечения

Для создания базы данных с требуемыми таблицами был написан скрипт, доступный в Приложении и на GitHub (URI - [https://github.com/vladcto/suai-labs/blob/bfad7c907f880feb8549dbc8078c4e9cfa2ced0d/6\\_semester/%D0%9C%D0%A1%D0%9F%D0%98%D0%A1%D0%A2/1/init.sql](https://github.com/vladcto/suai-labs/blob/bfad7c907f880feb8549dbc8078c4e9cfa2ced0d/6_semester/%D0%9C%D0%A1%D0%9F%D0%98%D0%A1%D0%A2/1/init.sql)).

Этот скрипт представляет собой SQL-скрипт для создания базы данных с названием `conference_db_lab1` и несколькими таблицами, предназначенными для хранения информации о студентах, университетах, факультетах, группах, конференциях и программе конференции.

Первым шагом скрипт создает базу данных `conference_db_lab1`, если она еще не существует, а затем переключается на использование этой базы данных. Затем он создает таблицу `university` для хранения данных о университетах, таблицу `faculty` для информации о факультетах, `uni_group` для хранения данных о группах, и `student` для информации о студентах.

Далее скрипт создает таблицу `conference` для хранения данных о научных конференциях, а также таблицу `topic` для хранения тем, связанных с конференциями, и информации об авторах этих тем. Наконец, создается

таблица `conference_program` для хранения программы конференции, включающей информацию о времени начала и окончания каждой темы, а также связи с соответствующей конференцией и темой.

Все созданные таблицы содержат необходимые внешние ключи для поддержки целостности данных между ними, что обеспечивает связность информации в базе данных. Диаграмма БД представлена на рисунке 2.

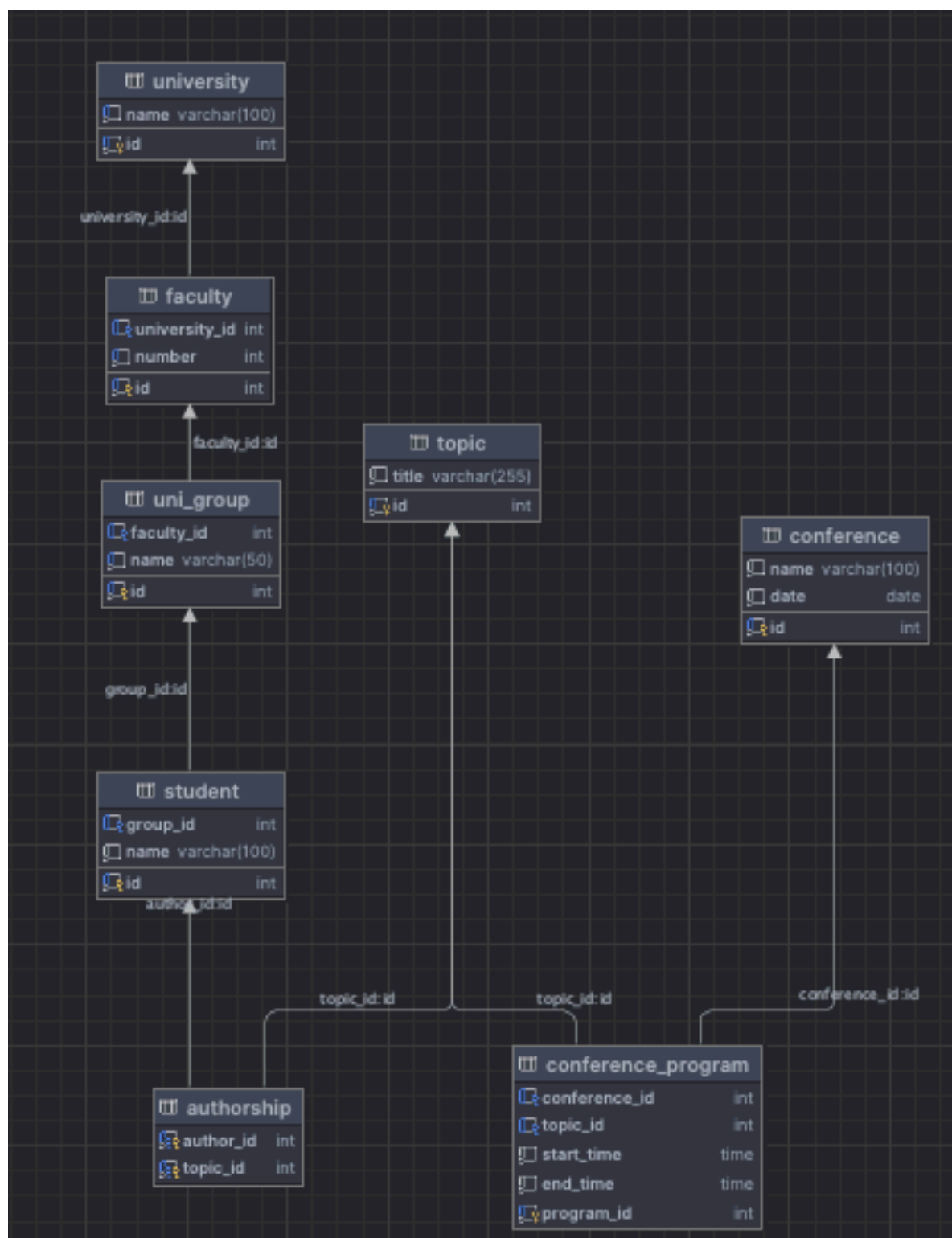


Рисунок 2 - Схема БД

### **3 Вывод**

В результате выполнения лабораторной работы мы успешно освоили навыки создания баз данных и таблиц, а также осуществили настройку правил проверки значений и установку ограничений ссылочной целостности.

Мы начали с разработки схемы базы данных, тщательно продумывая структуру для эффективного хранения сведений о ВУЗе, студентах, группах, факультетах, конференциях, темах докладов и программах конференций. С использованием языка SQL мы создали соответствующие таблицы, применив необходимые ограничения. Описывая структуру таблиц, мы определили ограничения на значения данных, гарантируя их корректность и актуальность.

Таким образом, лабораторная работа не только позволила нам успешно реализовать поставленные задачи, но и значительно улучшила наши навыки в области проектирования баз данных. Полученный опыт будет полезен при работе с реальными проектами, где эффективная организация данных играет ключевую роль.

## Приложение

```
1 CREATE DATABASE IF NOT EXISTS conference_db_lab1;
2
3 USE conference_db_lab1;
4
5 CREATE TABLE IF NOT EXISTS university
6 (
7     id      INT PRIMARY KEY AUTO_INCREMENT,
8     name    VARCHAR(100) NOT NULL UNIQUE
9 );
10
11 CREATE TABLE IF NOT EXISTS faculty
12 (
13     id              INT PRIMARY KEY AUTO_INCREMENT,
14     university_id   INT NOT NULL,
15     number          INT NOT NULL UNIQUE,
16     FOREIGN KEY (university_id) REFERENCES university (id)
17 );
18
19 CREATE TABLE IF NOT EXISTS uni_group
20 (
21     id              INT PRIMARY KEY AUTO_INCREMENT,
22     faculty_id      INT NOT NULL,
23     name            VARCHAR(50) NOT NULL UNIQUE,
24     FOREIGN KEY (faculty_id) REFERENCES faculty (id)
25 );
26
27 CREATE TABLE IF NOT EXISTS student
28 (
29     id              INT PRIMARY KEY AUTO_INCREMENT,
30     group_id        INT NOT NULL,
31     name            VARCHAR(100) NOT NULL,
32     CONSTRAINT fk_group_id FOREIGN KEY (group_id) REFERENCES
33         uni_group (id)
34 );
35
36 CREATE TABLE IF NOT EXISTS conference
37 (
38     id      INT PRIMARY KEY AUTO_INCREMENT,
39     name    VARCHAR(100) NOT NULL DEFAULT 'Научная конференция',
40     date    DATE NOT NULL
```



```

40 );
41
42 CREATE TABLE IF NOT EXISTS topic
43 (
44     id      INT PRIMARY KEY AUTO_INCREMENT,
45     title   VARCHAR(255) NOT NULL
46 );
47
48 CREATE TABLE IF NOT EXISTS authorship
49 (
50     author_id INT NOT NULL,
51     topic_id  INT NOT NULL,
52     CONSTRAINT pk_authorship PRIMARY KEY (author_id, topic_id),
53     CONSTRAINT fk_author_id FOREIGN KEY (author_id)
54         REFERENCES student (id),
55     CONSTRAINT fk_topic_id FOREIGN KEY (topic_id) REFERENCES
56         topic (id)
57 );
58
59 CREATE TABLE IF NOT EXISTS conference_program
60 (
61     program_id      INT PRIMARY KEY AUTO_INCREMENT,
62     conference_id   INT NOT NULL,
63     topic_id        INT NOT NULL,
64     start_time      TIME NOT NULL,
65     end_time        TIME NOT NULL,
66     CONSTRAINT fk_conf_id_program FOREIGN KEY (conference_id)
67         REFERENCES conference (id),
68     CONSTRAINT fk_topic_id_program FOREIGN KEY (topic_id)
69         REFERENCES topic (id),
70     CONSTRAINT check_time CHECK (start_time < end_time)
71 );

```