Министерство образования и науки РФ

Пермский национальный исследовательский политехнический университет

Электротехнический факультет

Кафедра Информационные технологии и автоматизированные системы

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ

Лабораторная работа №5

Выполнил:

Студент группы РИС-23-2б

Камалетдинов М.В.

Проверил:

Доцент кафедры ИТАС

Петренко А. А.

г. Пермь – 2024

Оглавление

[Цель работы 3](#_Toc181785562)

[Задачи работы 4](#_Toc181785563)

[Этапы выполнения 5](#_Toc181785564)

[1. Создание новой таблицы 5](#_Toc181785565)

[2. Создание трех триггеров (добавление, обновление, удаление) 7](#_Toc181785566)

[3. Работа в phpmyadmin 9](#_Toc181785567)

[Заключение 13](#_Toc181785568)

[Список использованной литературы 14](#_Toc181785569)

# Цель работы

Создание и использование триггеров в СУБД.

# Задачи работы

1. Изучить понятие триггер;

2. Изучить и знать наизусть синтаксис создания триггера;

3. Создать триггеры (на добавление, изменение и удаление) в базе данных лабораторной работы № 5, создав в БД дополнительную таблицу LOG;

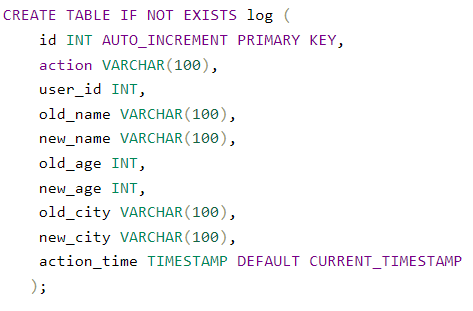
4. Отчет оформить, используя шаблон отчета.

# Этапы выполнения

## Создание новой таблицы

Для проверки работы тригерров будем использовать ранее созданную таблицу user в базе данных “kamaletdinov”, а также пропишем создание новой таблицы в базе данных “kamaletdinov”. Таблица “log”, которая будет хранить все изменения над таблицей “user”. Подробное описание создание таблицы с помощью SQL запроса:

1. CREATE TABLE IF NOT EXISTS LOG: Эта команда создает новую таблицу с именем LOG, но только в том случае, если таблица с таким именем еще не существует в базе данных. Это предотвращает ошибку, если попытаться создать таблицу, которая уже есть.
2. id INT AUTO\_INCREMENT PRIMARY KEY:
   * id: это поле будет использоваться как уникальный идентификатор для каждой записи в таблице.
   * AUTO\_INCREMENT: Значение этого поля будет автоматически увеличиваться на 1 при добавлении каждой новой записи, что обеспечивает уникальность идентификатора.
   * PRIMARY KEY: указывает, что это поле является первичным ключом таблицы, то есть оно должно быть уникальным и не может содержать NULL-значения.
3. action VARCHAR (100):
   * Это поле будет хранить строку (текст) длиной до 100 символов, представляющую действие, которое было выполнено (например, "создание", "изменение", "удаление").
4. user\_id INT:
   * Это поле будет хранить целое число, представляющее идентификатор пользователя, который выполнил действие.
5. old\_name VARCHAR (100):
   * Это поле будет хранить старое имя (например, имя пользователя или объекта), длиной до 100 символов.
6. new\_name VARCHAR (100):
   * Это поле будет хранить новое имя, длиной до 100 символов.
7. old\_age INT:
   * Это поле будет хранить старый возраст пользователя
8. new\_age
   * Это поле будет хранить новый возраст пользователя
9. old\_city VARCHAR (100):
   * Это поле будет хранить старый адрес город, длиной до 100 символов.
10. new\_city VARCHAR (100):
    * Это поле будет хранить новый город, длиной до 100 символов.
11. action\_time TIMESTAMP DEFAULT CURRENTTIMESTAMP:
    * Это поле будет хранить временную метку, когда действие было выполнено. По умолчанию оно будет заполняться текущей датой и временем в момент вставки записи.



*Рисунок 1 – Код программы, создание таблицы “log”*

## Создание трех триггеров (добавление, обновление, удаление)

Далее пропишем SQL запросы для создания триггеров. Всего будет три SQL запроса на создание трех триггеров: “after\_user\_insert”, “after\_user\_update”, “after\_user\_delete”.

“after\_user\_insert” - триггер, который выполняется после того, как в таблицу “user” добавили данные. И при добавлении данных в таблицу “user”, в таблицу log, записываются изменения таблицы “user”. А именно действие(action), новый айди (NEW.id), новое добавленное имя(NEW.name), новый добавленный возраст (NEW.age) и новый добавленный город (NEW.city).

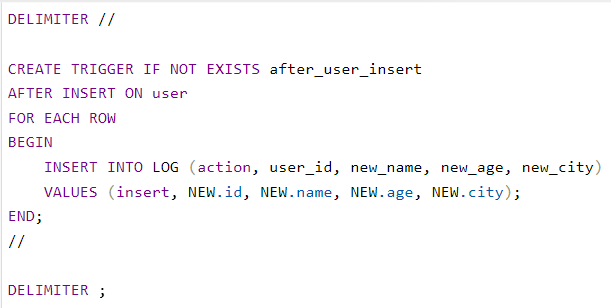


Рисунок 2 – Код программы, SQl запрос для создания триггера на добавление

“after\_user\_update” - триггер, который выполняется после того, как в таблице “user” изменили данные. И при изменении данных в таблице “users”, в таблицу log, записываются изменения таблицы “user”. А именно действие (action), айди в таблице “user” (user\_id), прошлое имя (OLD.name), новое имя (NEW.name), старый возраст (OLD.age), новый возраст (NEW.age), старый город (OLD.age), новый город (NEW.age).

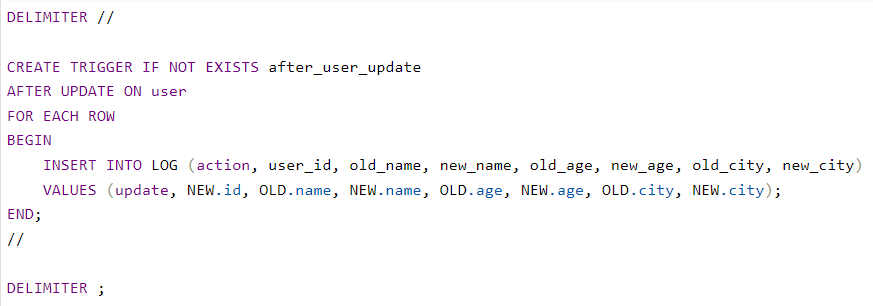


Рисунок 3 – Код программы, SQl запрос для создания триггера на обновление

“after\_user\_delete” - триггер, который выполняется после того, как из таблицы “user” удалили данные. И при удалении данных из таблицы “user”, в таблицу log, записываются изменения таблицы “user”. А именно действие (action), айди, удаляемых данных (OLD.id), удаленное имя (OLD.name), удаленный возраст (OLD.age) и удаленный город (OLD.city).

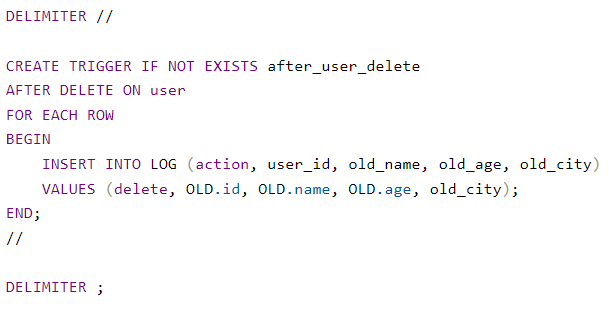


Рисунок 4 – Код программы, SQl запрос для создания триггера на удаление

## Работа в phpmyadmin

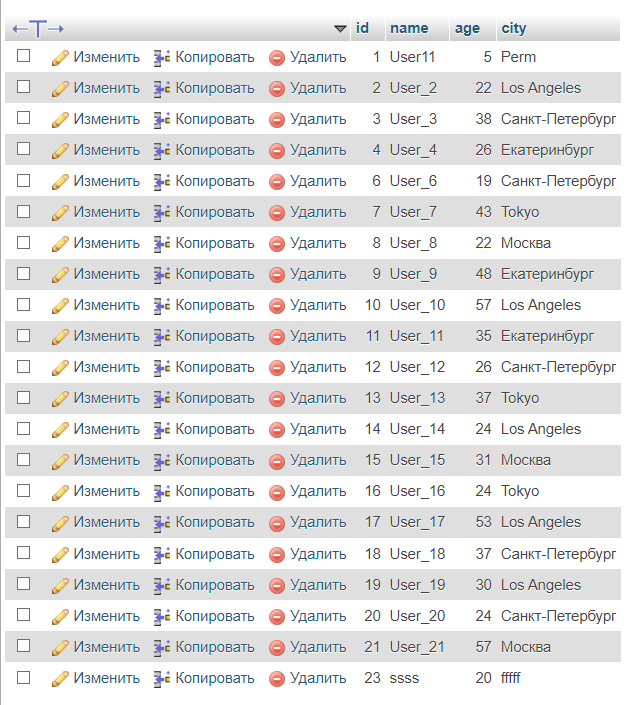


Рисунок 5 – Ранее созданная таблица user

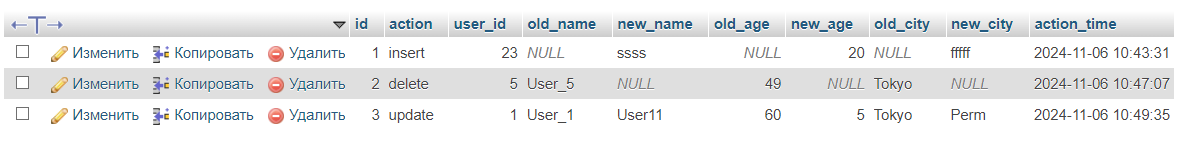


Рисунок 6 – Новая таблица log

Добавим строку в таблицу “user”, с помощью SQL запроса



Рисунок 7 – SQL запрос на добавление новой записи в таблицу

После этого в таблице log, появится новая запись.

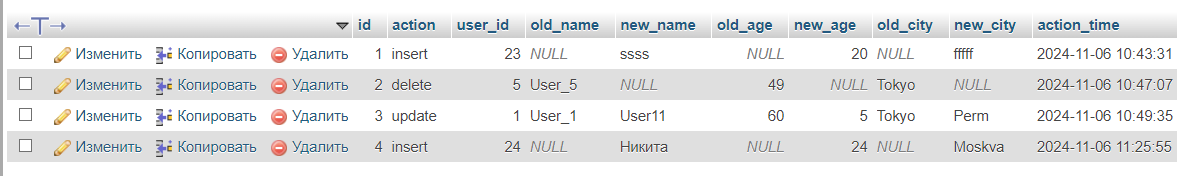


Рисунок 8 – Измененная таблица log

Теперь изменим данные в строке с id = 1, в таблице “user”, с помощью SQL запроса:



Рисунок 9 – SQL запрос на обновление записи в таблице

После этого в таблице log появится новая запись, а в таблице user изменится запись с id=1.

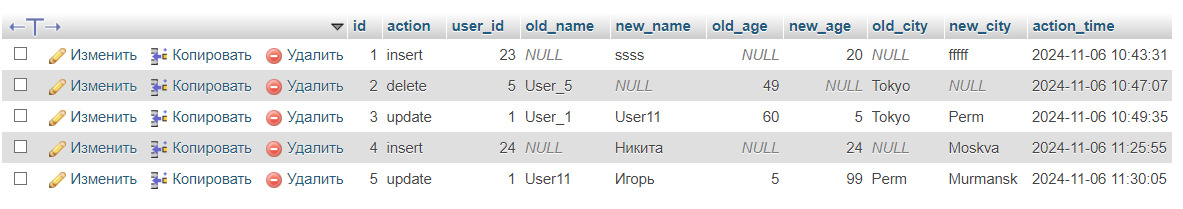


Рисунок 10 – Измененная таблица log

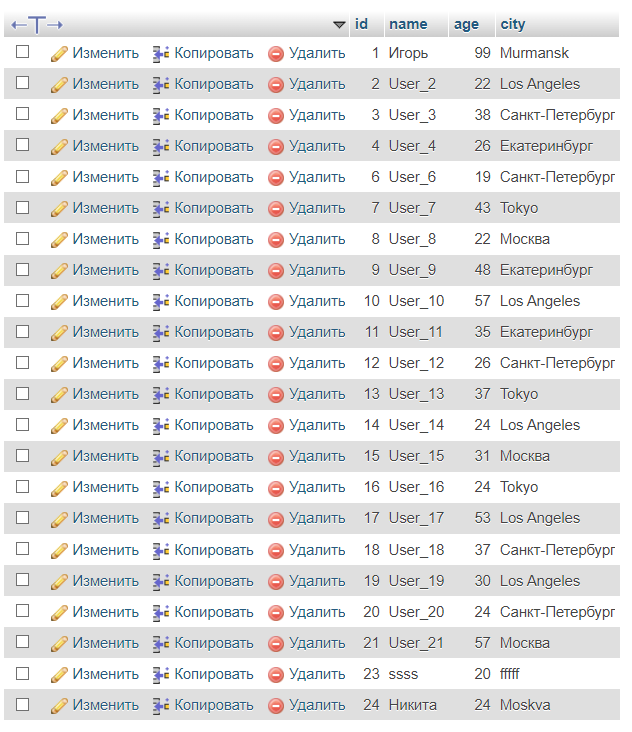


Рисунок 11 – Таблица user, с измененной записью id=1

Удалим строку из таблицы “user”, с помощью SQL запроса:



Рисунок 12 – SQL запрос на удаление записи в таблице

После этого из таблицы log, удалится запись с id=1.

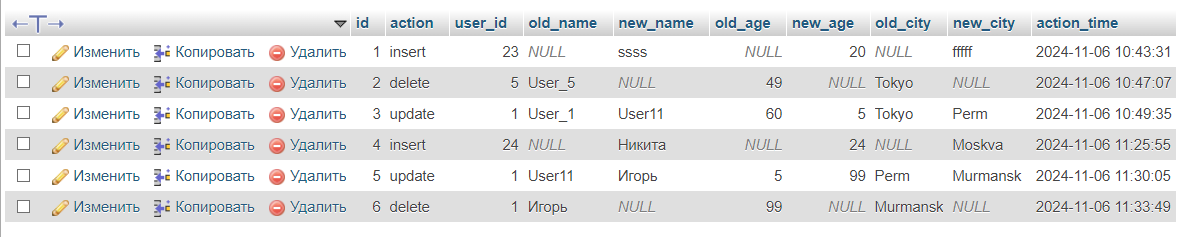


Рисунок 13 – Измененная таблица log

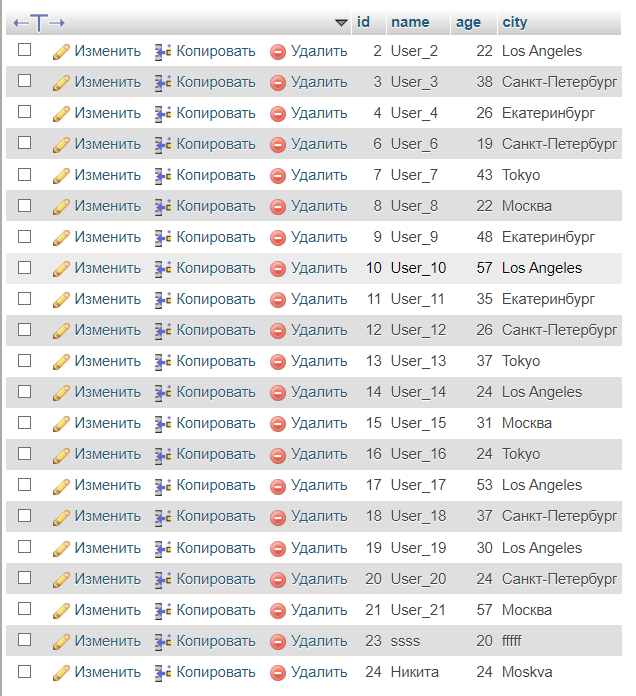


Рисунок 14 – Таблица user, с удаленной записью id=1

# Заключение

В ходе лабораторной работы была успешно выполнена серия задач, направленных на изучение и применение команд языка SQL, а также на разработку веб-страницы для визуализации результатов.

Было подробно изучено понятие триггера, создание триггера (на добавление, изменение и удаление).

На основе полученных знаний студент разработал веб-страницу, на которой были продемонстрированы данные таблицы LOG, в которой хранились данные об изменениях таблицы “users”.

В завершение работы, были выведены результаты выполнения команд на страницу браузера. Это не только продемонстрировало успешность выполнения поставленных задач, но и показало практическую значимость изученных команд для работы с базами данных. В итоге, лабораторная работа способствовала не только теоретическому, но и практическому освоению необходимых навыков в области управления базами данных.

# Список использованной литературы

1. <https://www.mysql.com>
2. <https://metanit.com/sql/mysql/>
3. <https://ospanel.io/docs/>
4. https://learn.microsoft.com/ru-ru/sql/t-sql/statements/create-trigger-transact-sql?view=sql-server-ver16