**Отчет по самостоятельной работе №**3

**по дисциплине МДК 01.02 “Технология разработки программного обеспечения”.**

Выполнил: студент

группы 319

Меркушин Евгений Романович

Дата 19.11.2024

**Цель работы**

Цель работы – научиться работать с MySQL, выполнять основные SQL-запросы для создания и работы с таблицами и данными.

**Основная структура задания**

1. Используйте следующую команду:
2. sudo apt-get update

sudo apt-get install mysql-server

1. **Настройка root-доступа к базе данных**:После установки MySQL необходимо настроить root-пользователя. По умолчанию, начиная с версии MySQL 5.7, используется аутентификация с использованием плагина auth\_socket, который позволяет подключаться к базе без пароля, если пользователь имеет соответствующие права в системе.Чтобы изменить это поведение и задать пароль для пользователя root, выполните следующие команды:Войдите в консоль MySQL:

sudo mysql

Измените метод аутентификации для пользователя root:

ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql\_native\_password BY 'newpassword';

FLUSH PRIVILEGES;

Замените newpassword на желаемый пароль.Теперь вы можете выйти из консоли MySQL и войти снова, используя новый пароль:

exit

mysql -u root -p

Enter password: [введите новый пароль]

# Задание 2: Создание базы данных

1. **Создание новой базы данных с помощью SQL-запроса**:Внутри консоли MySQL выполните следующий запрос:
2. CREATE DATABASE mydatabase;

USE mydatabase; -- переключаемся на созданную базу данных

1. **Создайте несколько таблиц и задайте их структуру**:Пример создания двух таблиц users и orders:Таблица users:
2. CREATE TABLE users (
3. id INT AUTO\_INCREMENT PRIMARY KEY,
4. name VARCHAR(255),
5. email VARCHAR(255) UNIQUE,
6. created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

Таблица orders:

CREATE TABLE orders (

order\_id INT AUTO\_INCREMENT PRIMARY\_KEY,

user\_id INT,

product\_name VARCHAR(255),

quantity INT,

total\_price DECIMAL(10, 2),

FOREIGN KEY (user\_id) REFERENCES users(id)

);

# Задание 3: Вставка данных

1. **Добавьте записи в таблицы с помощью SQL-запросов**:Вставим несколько записей в таблицу users и связанную с ней таблицу orders.Записи в таблице users:
2. INSERT INTO users (name, email) VALUES ('Иван Иванов', 'ivan@example.com');

INSERT INTO users (name, email) VALUES ('Петр Петров', 'petr@example.com');

Проверка вставленных данных:

SELECT \* FROM users;

Вы получите примерно такой результат:

Добавляем записи в таблицу orders, связанные с пользователями:

INSERT INTO orders (user\_id, product\_name, quantity, total\_price)

VALUES (1, 'Товар 1', 2, 200.00);

INSERT INTO orders (user\_id, product\_name, quantity, total\_price)

VALUES (2, 'Товар 2', 3, 300.00);

Проверяем вставленные данные:

SELECT \* FROM orders;

# Задание 4: Извлечение данных

1. **Напишите SQL-запросы для выборки данных с разными условиями**:
   * Выборка всех заказов конкретного пользователя:
   * SELECT o.order\_id, u.name, o.product\_name, o.quantity, o.total\_price
   * FROM orders o
   * INNER JOIN users u ON o.user\_id = u.id

WHERE u.id = 1;

* + Суммарная стоимость всех товаров, купленных каждым пользователем:
  + SELECT u.name, SUM(o.total\_price) AS total\_spent
  + FROM users u
  + LEFT JOIN orders o ON u.id = o.user\_id

GROUP BY u.id;

# Задание 5: Обновление и удаление данных

1. **Запросы для обновления и удаления данных**:Обновим цену товара у одного из заказов:
2. UPDATE orders SET total\_price = 250.00 WHERE order\_id = 1;

SELECT \* FROM orders WHERE order\_id = 1; -- проверка обновлённых данных

Удаление заказа:

DELETE FROM orders WHERE order\_id = 2;

``````sql

SELECT \* FROM orders; -- проверяем оставшиеся заказы

# Выводы работы

Таким образом, вы выполнили все задания, связанные с установкой, созданием баз данных, управлением данными и написанием различных SQL-запросов.