**Отчет по лабораторной работе № 1**

**Задание 1**

Раздел "Management":

Раздел "Management" предоставляет инструменты для администрирования и управления сервером MySQL. Основные функциональные группы включают:

Общая информация о сервере:

- Отображение основных параметров сервера: имя хоста, порт подключения, версию MySQL.

- Статус соединения с сервером и время его работы.

Конфигурация каталогов:

- Просмотр путей к основным каталогам сервера: директория данных, временные файлы, логи.

- Информация о расположении конфигурационных файлов.

Управление SSL-соединениями:

- Настройка и мониторинг параметров безопасного соединения.

- Конфигурация сертификатов и ключей для шифрования.

Мониторинг ресурсов системы:

- Отображение использования системных ресурсов: CPU, память, диск.

- Графическое представление загрузки ресурсов во времени.

Управление журналами:

- Просмотр и анализ лог-файлов сервера.

- Фильтрация и поиск событий в журналах ошибок и событий.

Настройка параметров сервера:

- Динамическое изменение глобальных и сессионных переменных.

- Мониторинг значений системных переменных и их состояния.

Управление резервными копиями:

- Создание и восстановление бэкапов баз данных.

- Планирование автоматических резервных копий.

Сервисное обслуживание:

- Инструменты для оптимизации таблиц и проверки их целостности.

- Репарация поврежденных таблиц при необходимости.

Раздел “Instance”: Раздел «Instance» предоставляет комплексные инструменты для управления экземпляром базы данных MySQL. В данном разделе можно выделить следующие ключевые функциональные группы:

Обзор состояния сервера:

- Отображение текущего статуса работы сервера MySQL, включая время его запуска и продолжительность работы.

- Информация о количестве активных подключений и их состоянии.

Настройки конфигурации:

- Возможность просмотра и изменения параметров конфигурационного файла сервера.

- Редактирование настроек производительности, таких как размер буферов, лимиты соединений и другие параметры.

Управление службами:

- Инструменты для запуска, остановки и перезапуска сервера MySQL.

- Просмотр журнала событий сервера для анализа ошибок и важных уведомлений.

Пользователи и привилегии:

- Администрирование пользователей базы данных: создание новых пользователей, изменение паролей, назначение прав доступа.

- Настройка ограничений для пользователей по IP-адресам или конкретным базам данных.

Бэкапы и восстановление:

- Создание резервных копий баз данных с использованием различных методов.

- Восстановление баз данных из ранее созданных резервных копий.

Схемы данных:

- Управление существующими схемами (базами данных): создание, удаление, модификация.

- Обзор содержимого схем, включая таблицы, представления, триггеры и хранимые процедуры.

Раздел “Performance”: Раздел «Performance» предназначен для мониторинга и оптимизации производительности базы данных MySQL. Он позволяет детально анализировать использование ресурсов и выявлять потенциальные проблемы.

Мониторинг ресурсов:

- Отображение информации о загрузке процессора, использования оперативной памяти и дисковых операций.

- Графики производительности для визуализации трендов и пиков нагрузки.

Статистика запросов:

- Анализ выполненных SQL-запросов, включая их частоту, длительность и объем обработанных данных.

- Выявление медленных запросов и запросов с высокой нагрузкой на систему.

Оптимизация индексов:

- Инструменты для анализа использования индексов таблиц.

- Рекомендации по добавлению или удалению индексов для повышения производительности запросов.

Кэширование:

- Мониторинг эффективности использования кэша запросов и буферов данных.

- Настраиваемые параметры для управления размером и поведением кэша.

Журналы производительности:

- Просмотр и анализ журналов производительности, где фиксируются различные события, связанные с работой сервера.

- Использование этих данных для долгосрочного планирования и оптимизации.

Аудит и отчеты:

- Создание отчетов о производительности за определенные периоды времени.

- Экспорт данных для дальнейшего анализа или представления руководству.

- Эти инструменты помогают администраторам баз данных своевременно реагировать на проблемы производительности и поддерживать оптимальную работу системы.

**Задание 2**

**Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.**

**Задание 3**

Созданная таблица:

CREATE TABLE `new\_table` (

`id` int NOT NULL AUTO\_INCREMENT,

`name` varchar(45) NOT NULL,

`email` varchar(45) NOT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `email\_UNIQUE` (`email`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;

**Задание 4**

После создания таблицы мы получаем такой SQL-запрос:

INSERT INTO `simple`.`new\_table` (`name`, `email`) VALUES ('Paul', 'paul@superpochta.com');

INSERT INTO `simple`.`new\_table` (`name`, `email`) VALUES ('Marina', 'marina.ivanova@hotmail.com');

INSERT INTO `simple`.`new\_table` (`name`, `email`) VALUES ('Ekaterina', 'ekaterina.petrova@outlook.com');

Мы получаем такой SQL-запрос после изменения ячейки «name»:

UPDATE `simpledb`.`new\_table` SET `name` = 'Bob' WHERE (`id` = '1');

**Задание 5.2**

Объяснение из документации: При формировании текущей даты/времени создавать строку не требу­ ется – следующие встроенные функции организуют доступ к систем­ ным часам и возвратят текущую дату и/или время в виде строки: CURRENT\_TIMESTAMP( );

Столбец, отслеживающий время последнего изменения пользовате­ лем определенной строки таблицы, использовал бы тип **timestamp** (временная метка). Этот тип содержит ту же информацию, что и тип datetime (год, месяц, день, час, минуту, секунду), но при добавлении или изменении строки таблицы сервер MySQL автоматически за­ полнит столбец **timestamp** текущими значениями даты/времени.

После построения таблицы получили:

ALTER TABLE `simpledb`.`new\_table`

ADD COLUMN `gender` ENUM('M', 'F') NULL AFTER `email`,

ADD COLUMN `bday` DATE NULL AFTER `gender`,

ADD COLUMN `postal` VARCHAR(10) NULL AFTER `bday`,

ADD COLUMN `rating` FLOAT NULL AFTER `postal`,

CHANGE COLUMN `id` `id` INT GENERATED ALWAYS AS () ;

**Задание 5.3**

Посмотрев таблицу, NULL могут иметь все значения, кроме id, так как оно явно указано NOT NULL. Поле created имеет значение по умолчанию CURRENT\_TIMESTAMP(), но если оно не было явно объявлено как NOT NULL, то технически оно тоже может быть NULL.

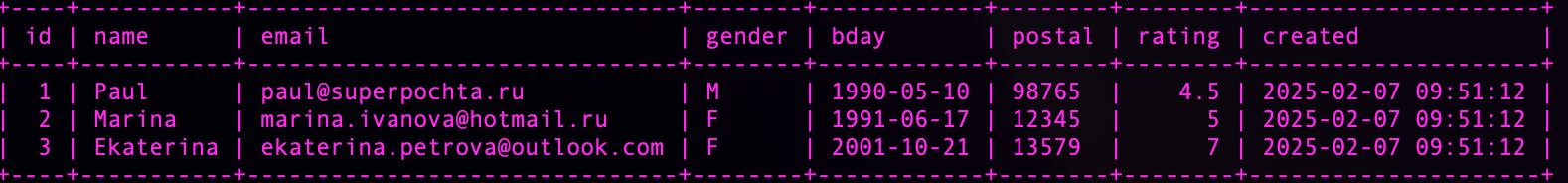
Я не мог создать таблицу в графическом интерфейсе, поэтому я создал таблицу через терминал:

Изображение выглядит как текст, Шрифт, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.

**Задание 6**

Мы добавили дополнительные данные о людях в нашей БД



**Задание 7**

Представленный файл содержит таблицу new\_table, в которой есть данные о трех людях.

SELECT – извлекает данные из таблицы

INSERT INTO – добавляет новую запись в таблицу

UPDATE – обновляет существующие записи в таблице

DELETE – удаляет записи из таблицы

CREATE TABLE – создает новую таблицу  
ALTER TABLE – изменяет структуру существующей таблицы

DROP TABLE – удаляет всю таблицу вместе со всеми ее данными

**Задание 8**

При удалении id из таблицы users удаляется вся строка с резюме в таблице resumes

Изображение выглядит как Шрифт, Графика, текст, графический дизайн

Контент, сгенерированный ИИ, может содержать ошибки.

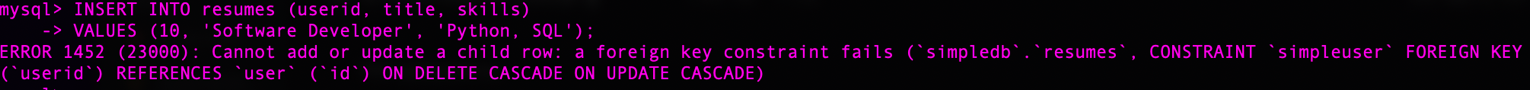
**Задание 9**

Можно создавать бесконечное количество резюме для каждого пользователя, просто буде генерироваться новый resumeid и каждое из них удалится, если удалить пользователя в таблице user.

Изображение выглядит как снимок экрана, фиолетовый, Фиолетовый, Красочность

Контент, сгенерированный ИИ, может содержать ошибки.

Невозможно добавить в таблицу resumes пользователя с id, которого не существует.



**Задание 10**

После удаления id пользователя, все резюме, которые относились к этому пользователю – были удалены.

Если изменить id пользователя, то он и изменится в таблицу resumes. Изображение выглядит как текст, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.

Изображение выглядит как снимок экрана, текст, фиолетовый

Контент, сгенерированный ИИ, может содержать ошибки.