**Задание 1.**

Раздел «Management»

2. Client Connections

Раздел Client Connections отображает список активных и неактивных клиентских подключений MySQL, добавляет возможность уничтожать операторы и подключения, а также просматривать дополнительные сведения и атрибуты подключения.

3. Users and Privileges

Вкладка Users and Privileges содержит список всех пользователей и привилегий, связанных с активным экземпляром сервера MySQL. На этой вкладке вы можете добавлять учетные записи пользователей и управлять ими, настраивать права и срок действия паролей.

* User Accounts: Состоит из вертикального поля, в котором перечислены все учетные записи пользователей, связанные с активным подключением MySQL. Список содержит имя каждого пользователя и имя хоста, на котором находится учетная запись.
* Login: Предоставляет ​​следующую информацию, относящуюся к выбранной учетной записи пользователя: Login Name, тип аутентификации, ограничение сопоставления хостов, пароль и подтверждение пароля.
* Account Limits: Определяет следующие ограничения для выбранной учетной записи пользователя:
  + Max. Queries: количество запросов, которые учетная запись может выполнить в течение одного часа.
  + Max. Updates: количество обновлений, которые учетная запись может выполнить в течение одного часа.
  + Max. Connections: сколько раз учетная запись может подключаться к серверу в течение часа.
  + Concurrent Connections: количество одновременных подключений к серверу, которое может иметь учетная запись.

- Administrative Roles: Роли - это быстрый способ предоставить пользователю набор привилегий в зависимости от работы, которую пользователь должен выполнять на сервере. Также можно назначить несколько ролей учетной записи пользователя или назначить привилегии непосредственно учетной записи без предварительного назначения ролей.

- Schema Privileges: уточняет способ назначения прав доступа к одной или нескольким схемам по учетной записи пользователя.

4. Status and System variables

Указывает все серверные переменные для соединения MySQL. Они разделены на переменные состояния и системные переменные. Эта вкладка позволяет при необходимости скопировать все переменные в буфер обмена.

5. Data Export

Эта вкладка позволяет экспортировать данные MySQL. Выберите схему, которую вы хотите экспортировать, при необходимости выберите конкретные объекты / таблицы схемы и сгенерируйте экспорт.

6. Data Import/Restore

Восстановите экспортированные данные из операции Data Export или из других данных, экспортированных с помощью команды mysqldump.

Раздел «Instance»

1. Startup / Shutdown: позволяет выполнять следующие действия по управлению службами:
   * Просматривать журнал сообщений о запуске.
   * Запустить и выключить сервер MySQL.
   * Просмотр текущего статуса сервера MySQL.
2. Server Logs: Вкладка отображает информацию журнала для сервера MySQL, представленного каждой вкладкой подключения. Для каждого подключения вкладка включает дополнительные разделы для общих журналов ошибок и журналов медленных запросов(если они есть).
3. Options File: Редактор файла параметров используется для просмотра и редактирования файла конфигурации MySQL путем установки флажков и других элементов управления графическим интерфейсом пользователя, а затем внесения изменений. MySQL Workbench делит файл параметров на отдельные группы в виде набора вкладок (General, Logging и т. д.)

Раздел «Performance»

1. Dashboard: Просматривайте статистику производительности сервера, представленную графически.
2. Performance Reports: Отчеты о производительности, которые могут помочь вам быстро выявить уязвимые места в ваших базах данных. Отчеты сгруппированы раскрываемым / сворачиваемым заголовкам.
3. Performance Schema Reports: Обеспечивает анализ операций сервера MySQL с помощью полезных отчетов высокого уровня.

**Задания 2 и 3.**

CREATE TABLE `simpledb`.`users` (

`id` INT NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(45) NOT NULL,

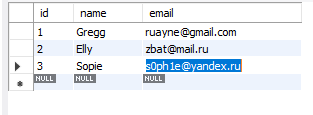
`email` VARCHAR(45) NOT NULL,

PRIMARY KEY (`id`),

UNIQUE INDEX `email\_UNIQUE` (`email` ASC) VISIBLE);

**Задание 4.**

1. SELECT \* FROM simpledb.users;



1. INSERT INTO `simpledb`.`users` (`id`, `name`, `email`) VALUES ('1', 'Gregg', 'ruayne@gmail.com');

INSERT INTO `simpledb`.`users` (`id`, `name`, `email`) VALUES ('2', 'Elly', 'zbat@mail.ru');

INSERT INTO `simpledb`.`users` (`id`, `name`, `email`) VALUES ('3', 'Sopie', 's0ph1e@yandex.ru');

1. UPDATE `simpledb`.`users` SET `name` = 'Zackary' WHERE (`id` = '1');

**Задание 5.**

1)

ALTER TABLE `simpledb`.`users`

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

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

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

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

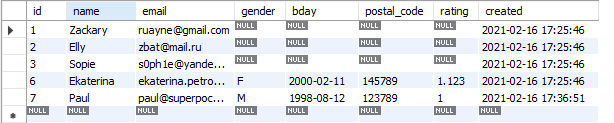
ADD COLUMN `created` TIMESTAMP NULL AFTER `rating`,

CHANGE COLUMN `name` `name` VARCHAR(50) NOT NULL ,

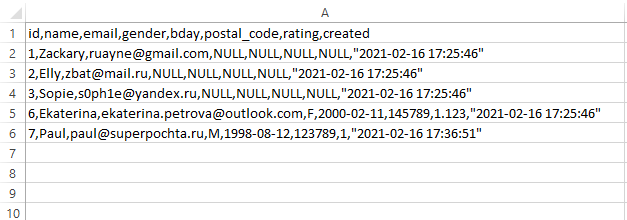
CHANGE COLUMN `email` `email` VARCHAR(50) NOT NULL ;

2) Все поля, кроме id, name, email и created принимают значения 0, сохраняют их при отсутствии изменений и не вызывают ошибок, если остаются пустыми.

**Задание 6.**



**Задание 7.**



**Задание 8.**

1)  
CREATE TABLE `simpledb`.`resume` (

`resumeid` INT NOT NULL AUTO\_INCREMENT,

`userid` INT NOT NULL,

`title` VARCHAR(100) NOT NULL,

`skills` TEXT NULL,

`created` TIMESTAMP NULL DEFAULT CURRENT\_TIMESTAMP(),

PRIMARY KEY (`resumeid`),

INDEX `userid\_idx` (`userid` ASC) VISIBLE,

CONSTRAINT `userid`

FOREIGN KEY (`userid`)

REFERENCES `simpledb`.`users` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE)

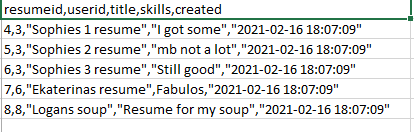
ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

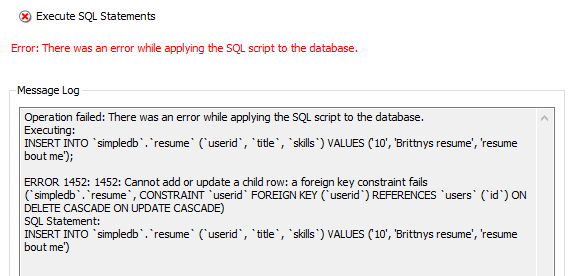
2) При изменении значения id у пользователя, связанное с этим пользователем id в таблице resume изменится. Если в таблице users удалить пользователя, все привязанные к нему резюме удалятся.

**Задание 9.**

1. У одного пользователя может быть 0 портфолио, или же любое количество, ограниченное только возможностями таблицы.





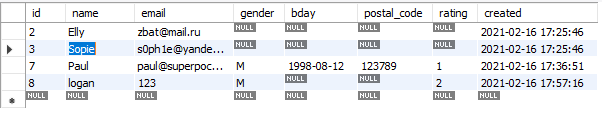


1. Не возможно создать резюме для человека, чьего id нет в таблице users.

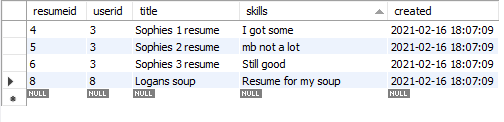
**Задание 10.**

1. DELETE FROM `simpledb`.`users` WHERE (`id` = '6');
2. Удаление человека из таблицы users повлекло за собой удаление резюме, связанных с id этого человека, из таблицы resume.
3. При изменении значения id у пользователя, связанное с этим пользователем id в таблице resume изменится.

a)



b)



c)



d)

