

Лабораторная работа №1

Задание 1

Раздел “Management”:

1. Раздел “Server Status”. В разделе отображается общая информация о сервере и подключении к нему. Информация логически сгруппирована. Можно выделить следующие группы:

- А) Общая информация (например, название хоста, номер порта, версия БД).
- Б) Настройки сервера (например, включен ли брандмауэр, используется ли SSL)
- В) Каталоги сервера
- Г) Сводка по используемым ресурсам компьютера (ОЗУ, процессор и т. д.)
- Д) Настройки соединения SSL (если SSL включена).

2. Раздел “ Client Connections”. Окно управления клиентскими подключениями включает в себя отображение сведений о подключениях к MySQL. Можно выделить следующие группы:

- А) Сведения(сведения о подключении, такие как идентификатор процесса, тип, пользователь, хост, инструментальный и дополнительная информация).
- Б) Блокировки(показывает информацию о блокировках).
- В) Атрибуты(атрибуты подключения, такие как ОС, имя клиента, версия клиента и платформа).

3. Раздел “Users and Privileges”. Раздел состоит из вертикального поля, в котором перечислены учетные записи пользователей поиска, связанные с активным подключением к MySQL. Список содержит имя каждого пользователя и имя хоста, на котором находится учетная запись. Используйте кнопки "Add Account", "Delete" и "Refresh" для управления списком учетных записей пользователей. При выборе учетной записи из списка предоставляются ее данные, которые отображаются в наборе параметров справа. Также присутствуют подразделы “Login”, “Account Limits”, “Administrative Roles” и “Schema Privileges”.

А) Вкладка "Login" предоставляет информацию, относящуюся к выбранной учетной записи пользователя (“Login name” - имя логина, “Authentication Type” - тип аутентификации, “Limit to Hosts Matching” - ограничение на совпадение хостов, “Password” и “Confirm password” – пароль и подтверждение пароля).

Б) Вкладка “Account Limits” учетной записи определяет ограничения для выбранной учетной записи пользователя (ограничения по количеству запросов, обновлений, количество одновременных и последовательных подключений к серверу, выполняемых в течение одного часа)

В) Роли – это быстрый способ предоставления пользователю набора привилегий, основанный на работе, которую пользователь должен выполнять на сервере. Также возможно назначить несколько ролей учетной записи пользователя или назначить привилегии непосредственно учетной записи без предварительного назначения ролей. После того как вы выберете роль для учетной записи пользователя, вы увидите накопленные привилегии на панели “Global Privileges”. Например, если вы выберете роль “BackupAdmin”, предоставленные привилегии включают “EVENT”, “LOCK RABLES”, “SELECT”, “SHOW DATABASES”. Полный список привилегий см. в разделе Привилегии, предоставляемые MySQL.

Г) Вкладка "Schema Privileges" уточняет способ назначения прав доступа к одной или нескольким схемам с помощью учетной записи пользователя.

4. Раздел "Status and Systems Variables". В разделе содержится список полного набора серверных переменных для активного подключения к MySQL. Можно установить одну или несколько глобальных системных переменных для сохранения при перезапусках сервера.

5. Разделы "Data Export" and "Data Import" Существует три способа экспорта и импорта данных в MySQL Workbench, каждый из которых служит для своей цели.

А) Мастер экспорта и импорта табличных данных (поддерживает операции импорта и экспорта с использованием файлов CSV и JSON и включает в себя несколько параметров конфигурации (разделители, выбор столбцов, выбор кодировки и многое другое). Мастер может выполняться на локальных или удаленно подключенных серверах MySQL, а действие импорта включает сопоставление таблиц, столбцов и типов)

Б) Мастер экспорта и импорта данных SQL (Используйте этот мастер для экспорта или импорта SQL, сгенерированного из MySQL Workbench или с помощью команды mysqldump)

В) Экспорт и импорт данных результатов (Экспортируйте или импортируйте данные непосредственно из или в результирующий набор. Операции экспорта и импорта, показанные в этом разделе, применяются только к данным результатов)

Раздел "Instance":

1. Раздел "Startup/Shutdown" позволяет выполнять следующие действия по управлению службами:

А) Просмотр журнала сообщений о запуске.

Б) Запуск и завершение работы экземпляра MySQL.

В) Просмотр текущего состояния экземпляра MySQL.

2. Раздел "Server Logs" отображает информацию журнала для сервера MySQL, представленную каждой вкладкой подключения. Для каждого подключения вкладка "Server Logs" включает дополнительные вкладки для общих журналов ошибок и журналов замедления (если таковые имеются).

3. Раздел "Options File" используется для просмотра и редактирования файла конфигурации MySQL путем установки флажков и других элементов управления графическим интерфейсом, а затем внесения изменений. MySQL Workbench делит файл параметров на свои собственные группы в виде набора вкладок (таких как Общие, Ведение журнала, InnoDB и так далее). Редактор файлов опций включает в себя следующие компоненты:

А) Опция группировки файлов, разделенная MySQL Workbench на удобные вкладки.

Б) Поле поиска Locate option для поиска в вашем файле конфигурации MySQL options.

В) Путь к файлу конфигурации, чтобы вы знали, какой файл конфигурации вы редактируете.

Раздел “Performance”:

1. Раздел “Dashboard”. В разделе можно посмотреть статистику производительности сервера на графической панели мониторинга.

A) Network Status. Здесь представлена статистика сетевого трафика, отправляемого и получаемого сервером MySQL по клиентским подключениям. Точки данных включают входящий сетевой трафик, исходящий сетевой трафик и клиентские подключения.

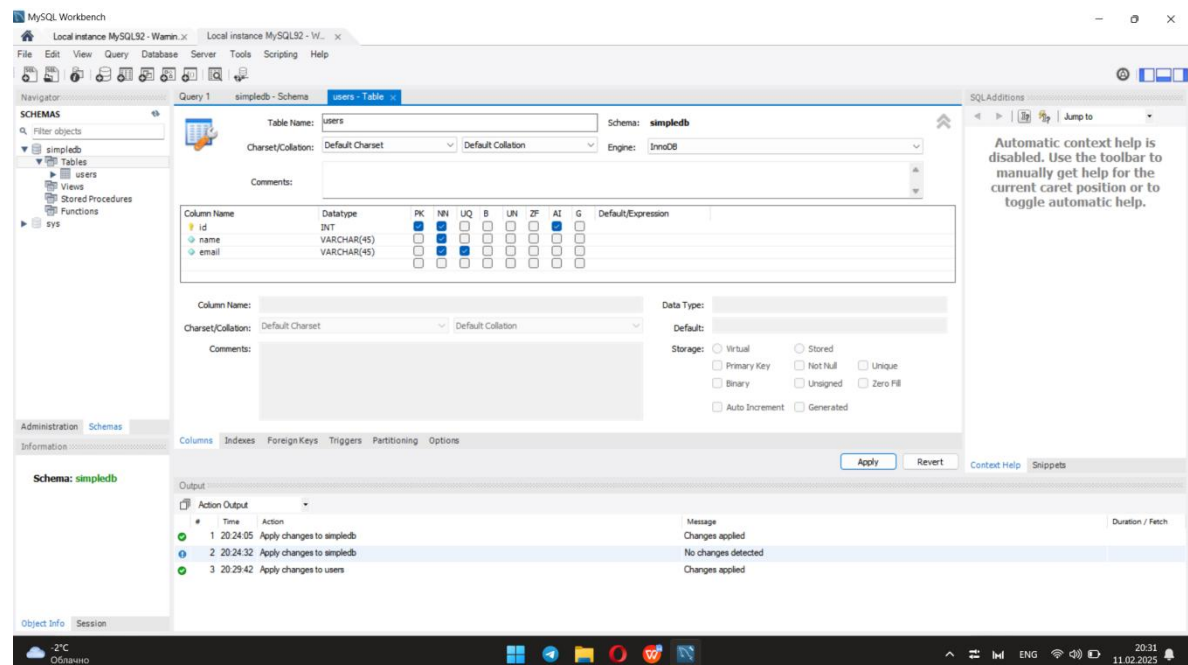
Б) MySQL Status. Показывает основную активность сервера MySQL и статистику производительности.

В) InnDB Status. Предоставляет обзор InnDB Buffer Pool и активности диска, которые генерируются механизмом InnDBstorage.

2. Раздел “Performance Reports”. Это графический интерфейс для конфигурирования и тонкой настройки инструментария схемы производительности.

3. Раздел “Performance Schema Setup”. Отчеты на основе схемы производительности дают представление о работе сервера MySQL с помощью полезных отчетов высокого уровня. MySQL Workbench использует Система просмотр схемы производительности для создания более 20 отчетов, помогающих анализировать производительность ваших баз данных MySQL. Отчеты помогают анализировать горячие точки ввода-вывода, обнаруживать дорогостоящие инструкции SQL и просматривать статистику ожидания и показатели движка InnoDB.

Задание 2

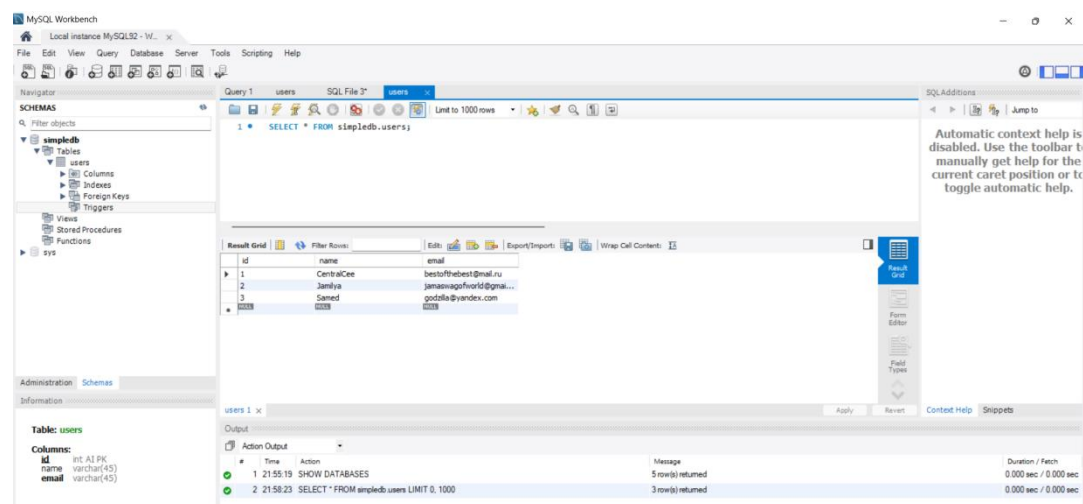


Задание 3

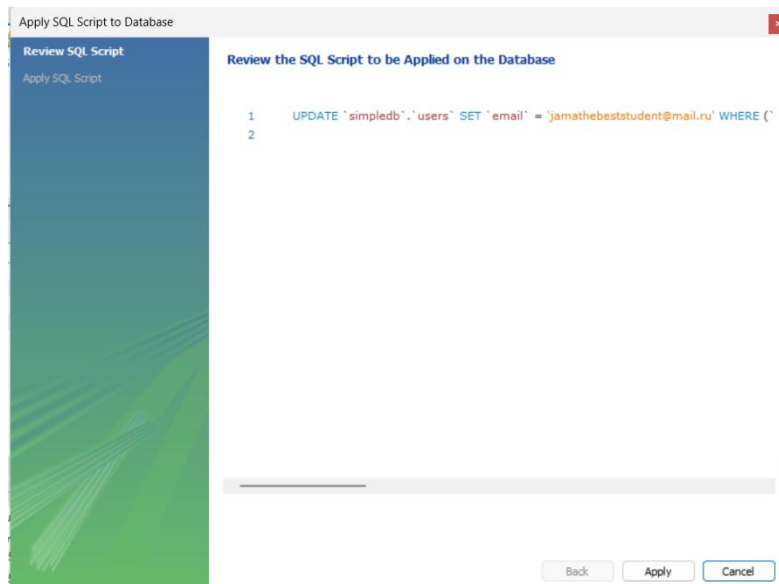
```
CREATE TABLE `users` (  
  `id` int NOT NULL AUTO INCREMENT,  
  `name` varchar(45) NOT NULL,  
  `email` varchar(45) NOT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE KEY `email` (`email`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;
```

Задание 4

```
CREATE TABLE `users` (  
  `id` int NOT NULL AUTO INCREMENT,  
  `name` varchar(45) NOT NULL,  
  `email` varchar(45) NOT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE KEY `email` (`email`)  
) ENGINE=InnoDB AUTO INCREMENT=4 DEFAULT CHARSET=utf8mb3;
```



Редактирование таблицы
Изменена почта



Задание 5



CURRENT_TIMESTAMP возвращает текущую дату и время в часовом поясе сеанса в значении типа данных TIMESTAMP (Хранит значение даты и времени в виде ГГГГ-ММ-ДД ЧЧ:ММ:СС. От 1970-01-01 00:00:01 до 2038-01-19 03:14:07) с указанием часового пояса

Поля, которые могут быть NULL:

name: Пользователь может не захотеть указывать своё имя.

email: Электронная почта также может оставаться необязательной для заполнения.

gender: Пол тоже является личной информацией, которую пользователь может предпочесть скрыть.

bday: Дата рождения – ещё одно поле, которое многие люди предпочитают оставлять незаполненным.

postal_code: Почтовый индекс может не указываться пользователем, особенно если он не связан с доставкой товаров или услуг.

rating: Рейтинг может отсутствовать до тех пор, пока система не начнёт его вычислять, поэтому это поле также может быть NULL.

Поля, которые должны быть NOT NULL:

id: Идентификатор записи всегда должен быть уникальным и ненулевым.

created: Время создания записи должно фиксироваться автоматически при добавлении новой строки, так что оно не может быть NULL.

Задание 6

Apply SQL Script to Database

Review SQL Script

Apply SQL Script

Review the SQL Script to be Applied on the Database

```
1 UPDATE 'simplifiedb`.`users` SET 'gender' = 'M', 'bday' = '1998-06-04', 'postal_code' = '193884', 'rating' = '6', 'created' = '16.02.25' WHERE ('id' = '1');
2 UPDATE 'simplifiedb`.`users` SET 'gender' = 'F', 'bday' = '2005-10-15', 'postal_code' = '23442', 'rating' = '3', 'created' = '16.02.25' WHERE ('id' = '2');
3 UPDATE 'simplifiedb`.`users` SET 'gender' = 'M', 'bday' = '2005-09-18', 'postal_code' = '235564', 'rating' = '2', 'created' = '16.02.25' WHERE ('id' = '3');
4 INSERT INTO 'simplifiedb`.`users` ('id', 'name', 'email', 'gender', 'bday', 'postal_code', 'rating', 'created') VALUES ('4', 'Madaline', 'nextgirl@google.com', 'F', '1998-06-24', '34555', '8', '16.02.25');
5
```

Back

Apply

Cancel

	id	name	email	gender	bday	postal_code	rating	created
▶	1	CentralCee	bestofthebest@mail.ru	M	1998-06-04	193884	6	2016-02-25 00:00:00
	2	Jamilya	jamathebeststudent@mail...	F	2005-10-15	23442	3	2016-02-25 00:00:00
	3	Samed	godzilla@yandex.com	M	2005-09-18	235564	2	2016-02-25 00:00:00
	4	Madaline	nextgirl@google.com	F	1998-06-24	34555	8	2016-02-25 00:00:00
	5	Paul	paul@superpochta.ru	M	1998-08-12	123789	1	NULL
	6	Ekaterina	ekaterina.petrova@outl...	F	2000-02-11	145789	1.123	NULL
✱	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Задание 7

```
/*
-- Query: SELECT * FROM simplifiedb.users
LIMIT 0, 1000
-- Date: 2025-02-16 11:25
*/
INSERT INTO `` ('id','name','email','gender','bday','postal_code','rating','created') VALUES (1,'CentralCee','bestofthebest@mail.ru','M','1998-06-04','193884',6,'2016-02-25 00:00:00');
INSERT INTO `` ('id','name','email','gender','bday','postal_code','rating','created') VALUES (2,'Jamilya','jamathebeststudent@mail.ru','F','2005-10-15','23442',3,'2016-02-25 00:00:00');
INSERT INTO `` ('id','name','email','gender','bday','postal_code','rating','created') VALUES (3,'Samed','godzilla@yandex.com','M','2005-09-18','235564',2,'2016-02-25 00:00:00');
INSERT INTO `` ('id','name','email','gender','bday','postal_code','rating','created') VALUES (4,'Madaline','nextgirl@google.com','F','1998-06-24','34555',8,'2016-02-25 00:00:00');
INSERT INTO `` ('id','name','email','gender','bday','postal_code','rating','created') VALUES (5,'Paul','paul@superpochta.ru','M','1998-08-12','123789',1,NULL);
INSERT INTO `` ('id','name','email','gender','bday','postal_code','rating','created') VALUES (6,'Ekaterina','ekaterina.petrova@outlook.com','F','2000-02-11','145789',1.123,NULL);
```

Между знаками “/*...*/” указана информация, из какой базы данных была взята информация, а также дата экспорта в данный файл.

Далее указаны заполненные поля таким образом, что после INSERT INTO указаны атрибуты, а после VALUES указаны наполняемое, соответствующее указанным атрибутам; на каждой новой строке указывается новая строка для самой таблицы.

Задание 8

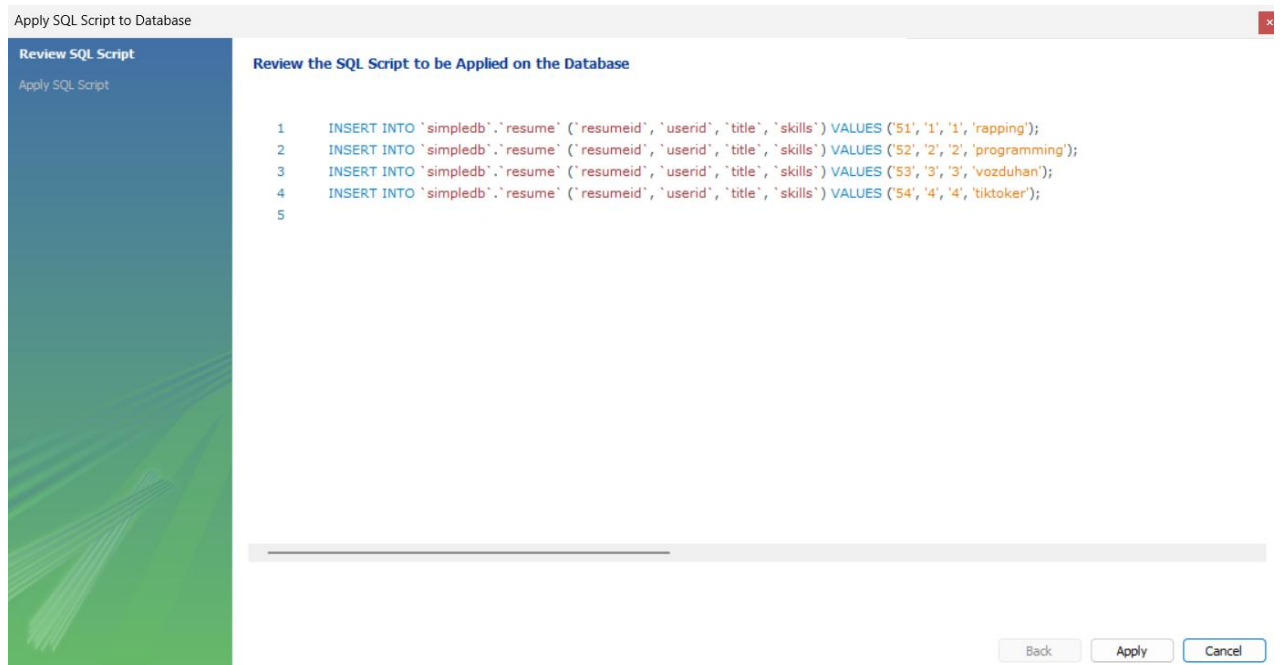


```
CREATE TABLE `simplifiedb`.`resume` (  
  `resumeid` INT NOT NULL AUTO_INCREMENT,  
  `userid` INT NOT NULL,  
  `title` VARCHAR(100) NOT NULL,  
  `skills` TEXT NULL,  
  `created` TIMESTAMP NULL DEFAULT CURRENT_TIMESTAMP(),  
  PRIMARY KEY (`resumeid`),  
  INDEX `userid_idx` (`userid` ASC) VISIBLE,  
  CONSTRAINT `userid`  
    FOREIGN KEY (`userid`)  
    REFERENCES `simplifiedb`.`users` (`id`)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE);
```

В данном SQL-запросе выполняется создание таблицы с некоторыми параметрами. Также указывается первичный ключ 'resumeid' и внешний ключ 'userid', который ссылается на таблицу "users", атрибуту 'id'.

Указаны параметр CASCADE при удалении или обновлении, что означает, что при выполнении данных действий в родительской таблице изменения автоматически произведутся в дочерних таблицах

Задание 9

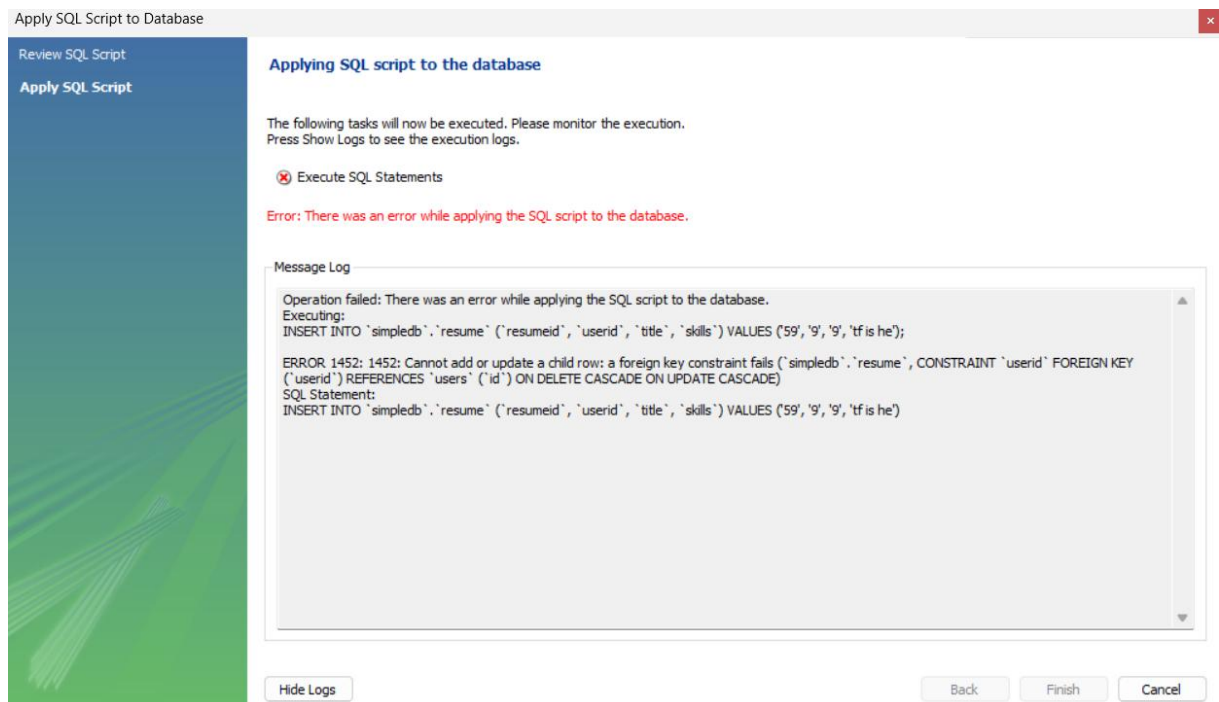


Минимальное количество резюме у одного пользователя - ноль

Максимальное количество резюме у одного пользователя может быть неограниченно

```
/*
-- Query: SELECT * FROM simplified.resume
LIMIT 0, 1000

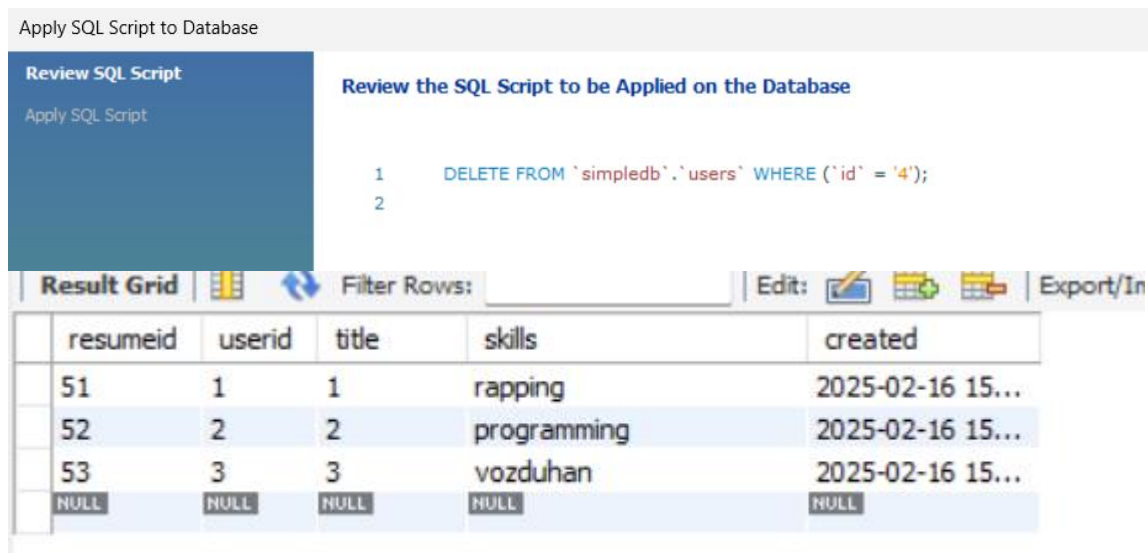
-- Date: 2025-02-16 15:48
*/
INSERT INTO `` (`resumeid`,`userid`,`title`,`skills`,`created`) VALUES (51,1,'1','rapping','2025-02-16 15:44:10');
INSERT INTO `` (`resumeid`,`userid`,`title`,`skills`,`created`) VALUES (52,2,'2','programming','2025-02-16 15:44:10');
INSERT INTO `` (`resumeid`,`userid`,`title`,`skills`,`created`) VALUES (53,3,'3','vozduhan','2025-02-16 15:44:10');
INSERT INTO `` (`resumeid`,`userid`,`title`,`skills`,`created`) VALUES (54,4,'4','tiktok','2025-02-16 15:44:10');
|
```

При попытке описания строки с несуществующем в “users” ‘id’ программа выдает ошибку.

Таким образом мы получили еще одну таблицу, в которой есть несколько данных о персонажах из прошлой таблицы. Кроме того, экспериментально выяснили, что добавить информацию во вторую таблицу о людях, неотмеченных в первой таблице невозможно.

Задание 10



При удалении пользователя Madaline из таблицы “users” удаляются его резюме (строки) в таблице резюме.

Review SQL Script

Apply SQL Script

Review the SQL Script to be Applied on the Database

```
1 UPDATE `simplified`.`users` SET `id` = '8' WHERE (`id` = '3');  
2
```

	resumeid	userid	title	skills	created
	51	1	1	rapping	2025-02-16 15...
	52	2	2	programming	2025-02-16 15...
	53	8	3	vozduhan	2025-02-16 15...
	NULL	NULL	NULL	NULL	NULL

При изменении 'id' в таблице "users" изменяет 'id' в таблице резюме