

Базы данных.

Лабораторная работа №1

Цель работы: исследовать на практике -
способы конструирования таблиц,
основные типы данных MySQL, режим
работы с БД с помощью визуального
конструктора (схемы данных), базовый
синтаксис запросов на языке SQL.

Автор: Чалапко Егор Витальевич, студент 2 курса ИВТ, 1 группы, 1 подгруппы

Задание 1. Используя справочные материалы по MySQL Workbench, опишите назначение пунктов меню Management (“Управление”), “Instance” (“Экземпляр БД”) и “Performance” (“Производительность”).

Задание 1

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:
 - o Max. Queries: количество запросов, которые учетная запись может выполнить в течение одного часа.
 - o Max. Updates: количество обновлений, которые учетная запись может выполнить в течение одного часа.
 - o Max. Connections: сколько раз учетная запись может подключаться к серверу в течение часа.
 - o 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:
 - o Просматривать журнал сообщений о запуске.
 - o Запустить и выключить сервер MySQL.
 - o 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.

Задание 2. Создать и настроить новую базу данных `simpledb`.

Задание 3. Скопируйте запрос, соответствующий созданию этой таблицы и вставьте его в отчет по выполнению этой лабораторной работы.

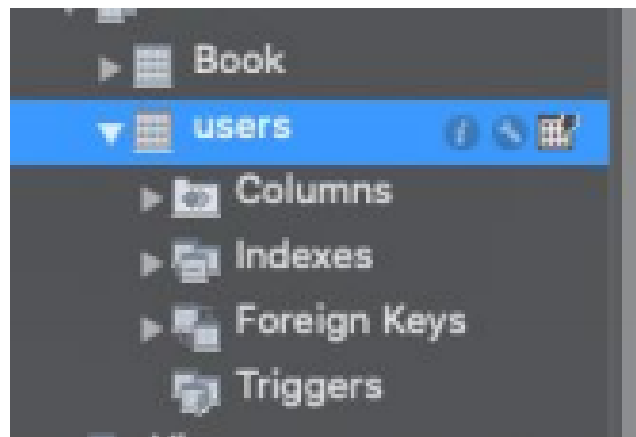
Задания 2 и 3

```
CREATE TABLE `simplifiedb`.`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

Добавьте данных в созданную таблицу. Для этого необходимо кликнуть на название таблицы левой кнопкой мыши и выбрать пиктограмму, где изображена таблица и молния (см. изображение ниже).

Какой SQL-запрос при этом выполнится? Отрадите его в отчете.



Далее, в нижней части экрана (где таблица) кликая по столбцам придумать и ввести 3 новые строчки, которые будут храниться в таблице.

Задание 4

1) `SELECT * FROM simpledb.users;`

2)

	id	name	email
	1	Gregg	ruayne@gmail.com
	2	Elly	zbat@mail.ru
▶	3	Sopie	s0ph1e@yandex.ru
•	NULL	NULL	NULL

3) `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');`

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

Задание 5

Дополните таблицу users так, чтобы получилась таблица со следующими полями и параметрами:

1. id int pk, not null
2. name varchar(50)
3. email varchar(45)
4. gender ENUM('M', 'F')
5. bday Date
6. postal_code varchar(10)
7. rating float
8. created TIMESTAMP DEFAULT CURRENT_TIMESTAMP()

Тип данных у поля created - TIMESTAMP, значение по умолчанию TIMESTAMP().

В отчете по ЛР опишите, что это означает, прочитав в документации или экспериментально определив.

Замечание 3

Определите самостоятельно какие поля могут быть NULL и будут иметь это значение в качестве значения по умолчанию, представив себе пользователя, который не очень хочет делиться персонализированной информацией.

Отразите это в отчёте.

Задание 5

1)

```
ALTER TABLE `simplifiedb`.`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

Дополните таблицу, добавив данные двумя способами:

- с помощью внесения данных вручную (как это было сделано ранее);
- с помощью выполнения SQL-запросов ниже;

```
INSERT INTO `simplifiedb`.`users` (`name`, `email`, `postal`, `gender`, `bday`, `rating`)  
VALUES ('Ekaterina', 'ekaterina.petrova@outlook.com', '145789', 'f', '2000-02-11',  
'1.123');
```

```
INSERT INTO `simplifiedb`.`users` (`name`, `email`, `postal`, `gender`, `bday`, `rating`)  
VALUES ('Paul', 'paul@superpochta.ru', '123789', 'm', '1998-08-12', '1');
```

Задание 6

[illegible]

Задание 7

С помощью кнопки “Export recordset to external file” сохраните файл с SQL-запросами добавления в таблицу данных, проанализируйте синтаксис этих запросов и вставьте их в отчет.

Задание 7

	A
1	id,name,email,gender,bday,postal_code,rating,created
2	1,Zackary,ruayne@gmail.com,NULL,NULL,NULL,NULL,"2021-02-16 17:25:46"
3	2,Elly,zbat@mail.ru,NULL,NULL,NULL,NULL,"2021-02-16 17:25:46"
4	3,Sopie,s0ph1e@yandex.ru,NULL,NULL,NULL,NULL,"2021-02-16 17:25:46"
5	6,Ekaterina,ekaterina.petrova@outlook.com,F,2000-02-11,145789,1.123,"2021-02-16 17:25:46"
6	7,Paul,paul@superpochta.ru,M,1998-08-12,123789,1,"2021-02-16 17:36:51"
7	
8	
9	
10	

Задание 8

Создайте еще одну таблицу с названием resume со следующей структурой:

- resumeid, INT, PK, NN, AI
- userid, INT, NN
- title, VARCHAR(100), NN
- skills, TEXT
- created, TIMESTAMP, *Default / Expression*: CURRENT_TIMESTAMP()

При конструировании внизу во вкладке Foreign Keys определите так называемый внешний ключ (foreign key), который будет определять связь между текущей таблицей resume и уже созданной таблицей user.

Введите в таблицу слева в столбец Foreign Key (внешний ключ): userid и определите таблицу, где они будут находиться: simpledb.users

В таблицу Foreign key details 'userid' щелкните мышкой рядом с полем userid так, чтобы оно было выделено и определите столбец-источник для значений - id.

Последний шаг: определить действия при операциях On Update и On Delete: для обоих выберите - Cascade.

Задание 8

1)

```
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)  
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

Наполните вторую таблицу данными так, чтобы в ней была информация хотя бы о нескольких резюме, связанных с уже существующими пользователями из таблицы users. Подумайте и напишите в отчете, сколько резюме может быть у одного пользователя (минимум и максимум)?

С помощью кнопки “Export recordset to external file” сохраните файл с SQL-запросами добавления в таблицу данных, проанализируйте синтаксис этих запросов и вставьте их в отчет.

Попробуйте добавить в таблицу resume строчку с userid несуществующего пользователя (такого пользователя id которого нет в таблице users).

Будет ли это возможно? Каков будет результат? Вставьте в отчет результат выполнения этого действия. Опишите полученный результат.

Задание 9

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

2)

resumeid	userid	title	skills	created
4	3	"Sophies 1 resume"	"I got some"	"2021-02-16 18:07:09"
5	3	"Sophies 2 resume"	"mb not a lot"	"2021-02-16 18:07:09"
6	3	"Sophies 3 resume"	"Still good"	"2021-02-16 18:07:09"
7	6	"Ekaterinas resume"	Fabulos	"2021-02-16 18:07:09"
8	8	"Logans soup"	"Resume for my soup"	"2021-02-16 18:07:09"

3)

 Execute SQL Statements

Error: There was an error while applying the SQL script to the database.

Message Log

```
Operation failed: There was an error while applying the SQL script to the database.  
Executing:  
INSERT INTO `simplifiedb`.`resume` (`userid`, `title`, `skills`) VALUES (10, 'Brittnys resume', 'resume  
bout me');  
  
ERROR 1452: 1452: Cannot add or update a child row: a foreign key constraint fails  
(`simplifiedb`.`resume`, CONSTRAINT `userid` FOREIGN KEY (`userid`) REFERENCES `users` (`id`) ON  
DELETE CASCADE ON UPDATE CASCADE)  
SQL Statement:  
INSERT INTO `simplifiedb`.`resume` (`userid`, `title`, `skills`) VALUES (10, 'Brittnys resume', 'resume  
bout me')
```

- 4) Не возможно создать резюме для человека, чьего id нет в таблице users.

Задание 10

Удалите одного или двух таких пользователей, что для них существуют записи в таблице `resume`.

Это можно сделать в режиме Result Grid или Form Editor аналогично тому как вносятся изменения в записи таблицы.

Вставьте в отчет SQL-запросы, которые вы увидите после нажатия Apply.

Что произойдет со связанными сущностями в таблице `resume`? Отрадите ответ с комментариями в отчете.

Что произойдет, если в таблице `users` будет изменен `id` какого-то существующего пользователя. Отрадите ответ и сделайте пояснение (скриншот) того, что произойдет.

Задание 10

- 1) DELETE FROM `simplifiedb`.`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)

	id	name	email	gender	bday	postal_code	rating	created
	2	Elly	zbat@mail.ru	NULL	NULL	NULL	NULL	2021-02-16 17:25:46
▶	3	Sophie	sophie@yande...	NULL	NULL	NULL	NULL	2021-02-16 17:25:46
	7	Paul	paul@superpoc...	M	1998-08-12	123789	1	2021-02-16 17:36:51
	8	logan	123	M	NULL	NULL	2	2021-02-16 17:57:16
•	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

b)

	resumeid	userid	title	skills	created
	4	3	Sophies 1 resume	I got some	2021-02-16 18:07:09
	5	3	Sophies 2 resume	mb not a lot	2021-02-16 18:07:09
	6	3	Sophies 3 resume	Still good	2021-02-16 18:07:09
▶	8	8	Logans soup	Resume for my soup	2021-02-16 18:07:09
•	NULL	NULL	NULL	NULL	NULL

c)

	15	Sophie	sophie@yande...	NULL	NULL	NULL	NULL	2021-02-16 17:25:46
•	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

d)

	4	15	Sophies 1 resume	I got some	2021-02-16 18:07:09
	5	15	Sophies 2 resume	mb not a lot	2021-02-16 18:07:09
	6	15	Sophies 3 resume	Still good	2021-02-16 18:07:09

Вывод:

В итоге, нам удалось изучить способы конструирования таблиц в MySQL, основные типы данных и режим работы с БД с помощью визуального конструктора, а также базовый синтаксис запросов на языке SQL.