

## Практическое задание по теме “CRUD — операции”

1. Повторить все действия по доработке БД vk.
2. Заполнить новые таблицы.
3. Повторить все действия CRUD.

Ниже представлены скрины экрана в редакторе Dbeaver с с командами CRUD операций

```
SHOW TABLES;

SELECT * FROM users;

-- add user photo
ALTER TABLE profiles ADD COLUMN photo_id INT UNSIGNED AFTER user_id;

DESC profiles;
-- add user status
CREATE TABLE user_statuses (
    id INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(150) NOT NULL UNIQUE );

INSERT user_statuses( id, name)
VALUES (1, 'active'), (2, 'blocked'), (3, 'deleted');

ALTER TABLE users ADD COLUMN status_id INT UNSIGNED NOT NULL DEFAULT 1 AFTER phone;
ALTER TABLE profiles ADD COLUMN is_private BOOLEAN DEFAULT FALSE AFTER country;
ALTER TABLE friendship ADD COLUMN declined_at DATETIME AFTER confirmed_at;
ALTER TABLE messages ADD COLUMN media_id INT UNSIGNED NOT NULL DEFAULT 1 AFTER body;
ALTER TABLE friendship DROP COLUMN requested_at;

SELECT * FROM users LIMIT 10;

UPDATE users SET updated_at = CURRENT_TIMESTAMP WHERE created_at > updated_at ;

SELECT * FROM profiles LIMIT 10;

UPDATE profiles SET photo_id = FLOOR(1 + RAND()*101);

CREATE TEMPORARY TABLE genders(name CHAR(1));
INSERT INTO genders VALUES ('m'), ('f');
SELECT * FROM genders;

UPDATE profiles SET gender = ( SELECT * FROM genders ORDER BY RAND() LIMIT 1);

-- CREATE TEMPORARY TABLE countries (name CHAR(50));
```

```

-- INSERT INTO countries VALUES ('Russian Federation'), ('Belarus'), ('Germany');
-- SELECT * FROM countries;

-- UPDATE profiles SET country = ( SELECT * FROM countries ORDER BY RAND() LIMIT 1);

UPDATE profiles SET is_private = TRUE WHERE photo_id > FLOOR(1 + RAND()*101) ;

SELECT * FROM messages;

UPDATE messages SET from_user_id = FLOOR(1 + RAND()*101),
    to_user_id = FLOOR(1 + RAND()*101),
    media_id = FLOOR(1 + RAND()*101);

SELECT * FROM media LIMIT 10;

UPDATE media SET user_id = FLOOR(1 + RAND()*101);

CREATE TEMPORARY TABLE extentions(name CHAR(20));
INSERT INTO extentions VALUES ('mp3'), ('avi'), ('mkv'), ('vob'), ('jpg');
SELECT * FROM extentions;

UPDATE media SET filename = CONCAT(
    'https://dropbox.net/vk/', filename, (SELECT * FROM extentions ORDER BY RAND() LIMIT 1));

UPDATE media SET `size` = FLOOR(100000 + RAND()*1000000) WHERE size < 1000;

SELECT CONCAT(
    '{ "owner": "',
    (SELECT CONCAT( first_name, ' ', last_name ) FROM users WHERE id = user_id),
    '"}' ) FROM media;

UPDATE media SET metadata = CONCAT(
    '{ "owner": "',
    (SELECT CONCAT( first_name, ' ', last_name ) FROM users WHERE id = user_id),
    '"}' );

ALTER TABLE media MODIFY COLUMN metadata JSON;

SELECT * FROM media_types;
-- DELETE FROM media_types;
TRUNCATE media_types;
INSERT media_types( id, name )
VALUES (1, 'photo'), (2, 'video'), (3, 'audio');

UPDATE media SET media_type_id = FLOOR(1 + RAND()*3);

SELECT * FROM friendship LIMIT 10;

TRUNCATE friendship_statuses;

INSERT INTO friendship_statuses (name) VALUES ('Requested'),('Confirmed'),('Rejected');

UPDATE friendship SET user_id = FLOOR(1 + RAND()*101),
    friend_id = FLOOR(1 + RAND()*101),
    status_id = FLOOR(1 + RAND()*3);

UPDATE friendship SET declined_at = (SELECT updated_at FROM users ORDER BY RAND() LIMIT 1);
UPDATE friendship SET friend_id = friend_id + 1 WHERE user_id = friend_id ;

SELECT * FROM communities;
DELETE FROM communities WHERE id > 20;

SELECT * FROM communities_users LIMIT 101;

UPDATE communities_users SET community_id = FLOOR(1 + RAND()*20), user_id = FLOOR(1 + RAND()*101).

```

#### 4. Подобрать сервис-образец для курсовой работы.

В качестве курсовой работы планируется разработать БД для обучающего ресурса.

Предполагается на сайте есть обучающие материалы. Пользователь регистрируется, знакомится с учебной программой, проходит тестирование. При этом собирается статистика о уровне и результатах пользователя. В каждом тесте происходит случайная выборка вопросов.

По мере проработки таблицы будут уточнены и наполнены кодом.

Приблизительный набор таблиц:

- users – пользователь. За основу возможно взять структуру рассмотренных примеров  
уровень — количество освоенного материала
- user\_profiles – профиль пользователя, За основу возможно взять структуру рассмотренных примеров
- questions - таблица вопросы содержит:  
раздел или тема,  
идентификатор вопроса,
- question – вопрос  
собственно вопрос  
по идее несколько вариантов ответов (минимум 2 максимум до 10)  
вариант ответа в виде изображения либо текста,  
ссылка на изображение или текст  
флаг правильности ответа  
идентификатор ответа
- user\_statistics – Статистика пользователя по прохождению тестирования отображает прогресс по усвоению материала.  
Раздел или тема,  
Идентификатор вопроса  
последний ответ пользователя  
число попыток,  
число правильных ответов