

Лабораторная работа 1

Выполнил Фролов А.А. 2-к., ИВТ-2

Описание функций

Раздел "Management"

Раздел "Server Status".

В разделе отображается общая информация о сервере и подключении к нему. Информация логически сгруппирована. Можно выделить следующие группы:

1. Общая информация (например, название хоста, номер порта, версия БД).
2. Настройки сервера (например, включен ли брандмауэр, используется ли SSL)
3. Каталоги сервера
4. Сводка по используемым ресурсам компьютера (ОЗУ, процессор и т. д.)
5. Настройки соединения SSL (если SSL включена).

Раздел "Client Connections".

В разделе отображается информация о подключениях в виде таблицы.

Раздел "Users and Privileges"

Раздел для настройки пользователей. В левой части список пользователей для выбора. В правой части окно для редактирования полномочий, который включает вкладки:

1. *Login* - данные авторизации
2. *Account Limits* - настройка максимально допустимых значений для работы с БД
3. *Administrative Roles* - права администратора
4. *Schema Privileges* - права для работы со схемами

Раздел "Status and System Variables"

Раздел с переменными. В левой части категории в правой части окно с переменными в выбранной категории.

Раздел "Data Export"

Раздел для экспорта данных. Есть две вкладки:

1. *Object selection* - для настроек экспорта
2. *Export progress* - для отслеживания процесса

Раздел "Data Import/Restore"

Раздел для импорта данных. Есть две вкладки:

1. *Import from Disk* - настройка импорта

2. *Import Progress* - отслеживание прогресса импорта

Раздел "Instance"

Раздел "Startup/Shutdown"

Окно отображающее статус сервера, позволяющее запускать и останавливать его.

Так же есть окно *Startup Message Log* с логами.

Раздел "Server Logs"

Раздел с логами.

1. *Error Log File* - вкладка с Error - логами

2. *Slow Log File* - вкладка с Slow - логами

Раздел "Options File"

Отображает всевозможные настройки. Есть такие вкладки настроек как:

1. Основные

2. Авторизация

3. InnoDB

4. Сетевые

5. Расширенные

6. Другие

7. Безопасность

8. Репликация

9. MyISAM

10. Производительность

Так же есть функция поиска нужной настройки.

Раздел "Perfomance"

Раздел "Dashboard"

Отображает различные графики статусов сети, MySQL, InnoDB

Раздел "Perfomance Reports"

Данные по различным пунктам производительности.

Раздел "Perfomance Schema Setup"

Работа с БД

Команда создания БД:

```
1 CREATE SCHEMA `simplesdb` DEFAULT CHARACTER SET utf8 ;
```

Команда создания таблицы:

```
1 CREATE TABLE `users` (
2     `id` int NOT NULL AUTO_INCREMENT,
3     `name` varchar(45) NOT NULL,
4     `email` varchar(45) NOT NULL,
5     PRIMARY KEY (`id`),
6     UNIQUE KEY `email_UNIQUE` (`email`)
7 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;
```

Команды добавления записей:

```
1 INSERT INTO `simpledb`.`users` (`name`, `email`) VALUES ('Petr',
'petr@mail.ru');
2 INSERT INTO `simpledb`.`users` (`name`, `email`) VALUES ('Ivan',
'ivan@mail.ru');
3 INSERT INTO `simpledb`.`users` (`name`, `email`) VALUES ('Alex',
'alex@mail.ru');
```

Обновление полей:

```
1 UPDATE `simpledb`.`users` SET `email` = 'petr_new@mail.ru' WHERE (`id` =
'1');
```

Дополнение таблицы:

```
1 ALTER TABLE `simpledb`.`users`
2 ADD COLUMN `gender` ENUM('M', 'F') NOT NULL AFTER `email`,
3 ADD COLUMN `bdy` DATE NULL DEFAULT NULL AFTER `gender`,
4 ADD COLUMN `postal_code` VARCHAR(10) NULL DEFAULT 'Null' AFTER `bdy`,
5 ADD COLUMN `rating` FLOAT NULL DEFAULT NULL AFTER `postal_code`,
6 ADD COLUMN `created` TIMESTAMP NULL DEFAULT CURRENT_TIMESTAMP() AFTER
`rating`,
7 CHANGE COLUMN `name` `name` VARCHAR(50) NOT NULL ;
```

CURRENT_TIMESTAMP() присваивает время в которое выполнен запрос

Содержимое файла по users:

```
1 /*
2 -- Query: SELECT * FROM simpledb.users
3 LIMIT 0, 1000
4
5 -- Date: 2026-02-05 11:20
6 */
7 INSERT INTO `simpledb`.`users`(`id`, `name`, `email`, `gender`, `bdy`, `postal_code`, `rating`, `created`)
VALUES (1, 'Petr', 'petr_new@mail.ru', 'M', NULL, 'Null', NULL, '2026-02-05
11:11:29');
```

```

8  INSERT INTO ``
(`id`, `name`, `email`, `gender`, `bday`, `postal_code`, `rating`, `created`)
VALUES (2, 'Ivan', 'ivan@mail.ru', 'M', NULL, 'Null', NULL, '2026-02-05
11:11:29');
9  INSERT INTO ``
(`id`, `name`, `email`, `gender`, `bday`, `postal_code`, `rating`, `created`)
VALUES (3, 'Alex', 'alex@mail.ru', 'M', NULL, 'Null', NULL, '2026-02-05
11:11:29');
10 INSERT INTO ``
(`id`, `name`, `email`, `gender`, `bday`, `postal_code`, `rating`, `created`)
VALUES (4, 'Fedya', 'fedor@mail.ru', 'M', '1994-06-10', '195786', 4.9, '2026-
02-05 11:15:52');
11 INSERT INTO ``
(`id`, `name`, `email`, `gender`, `bday`, `postal_code`, `rating`, `created`)
VALUES (5, 'Ekaterina', 'ekaterina.petrova@outlook.com', 'F', '2000-02-
11', '145789', 1.123, '2026-02-05 11:17:59');
12 INSERT INTO ``
(`id`, `name`, `email`, `gender`, `bday`, `postal_code`, `rating`, `created`)
VALUES (6, 'Paul', 'paul@superpochta.ru', 'M', '1998-08-
12', '123789', 1, '2026-02-05 11:18:20');

```

Содержит команды по заполнению каждой строки

Связывание таблиц ключом:

```

1  ALTER TABLE `simpledb`.`resume`
2  ADD INDEX `userid_idx` (`userid` ASC) VISIBLE;
3  ;
4  ALTER TABLE `simpledb`.`resume`
5  ADD CONSTRAINT `userid`
6    FOREIGN KEY (`userid`)
7    REFERENCES `simpledb`.`users` (`id`)
8    ON DELETE CASCADE
9    ON UPDATE CASCADE;

```

Содержимое файла по заполнению resume:

```

1  /*
2   -- Query: SELECT * FROM simpledb.resume
3   LIMIT 0, 1000
4
5   -- Date: 2026-02-05 15:15
6   */
7  INSERT INTO ``(`resumeid`, `userid`, `title`, `skills`, `created`) VALUES
(1, 1, 'resume1', 'all', '2026-02-05 15:14:21');
8  INSERT INTO ``(`resumeid`, `userid`, `title`, `skills`, `created`) VALUES
(2, 2, 'resume2', 'none', '2026-02-05 15:14:21');
9  INSERT INTO ``(`resumeid`, `userid`, `title`, `skills`, `created`) VALUES
(3, 1, 'resume1_2', 'all^2', '2026-02-05 15:14:42');

```

Невозможно добавить резюме с id несуществующего пользователя. Выдает ошибку по foreign key:

```

1 Operation failed: There was an error while applying the SQL script to
the database.
2 Executing:
3 INSERT INTO `simpledb`.`resume`(`userid`, `title`, `skills`) VALUES
('10', 'resume10', 'none');
4
5 ERROR 1452: 1452: Cannot add or update a child row: a foreign key
constraint fails (`simpledb`.`resume`, CONSTRAINT `userid` FOREIGN KEY
(`userid`) REFERENCES `users`(`id`) ON DELETE CASCADE ON UPDATE
CASCADE)
6 SQL Statement:
7 INSERT INTO `simpledb`.`resume`(`userid`, `title`, `skills`) VALUES
('10', 'resume10', 'none')
```

Удаление пользователя:

```
1 DELETE FROM `simpledb`.`users` WHERE (`id` = '2');
```

в таблице запись о резюме пользователя с id=2 удаляется

Смена id:

```
1 UPDATE `simpledb`.`users` SET `id` = '123' WHERE (`id` = '1');
```

Screenshot of MySQL Workbench showing a database schema and execution details.

Result Grid:

| resumeid | userid | title | skills | created |
|----------|--------|-----------|--------|---------------------|
| 1 | 123 | resume1 | all | 2026-02-05 15:14:21 |
| 3 | 123 | resume1_2 | all^2 | 2026-02-05 15:14:42 |
| * | NULL | NULL | NULL | NULL |

Output:

| Action Output | # | Time | Action | Message | Duration / Fetch |
|---------------|----|----------|--|---|-----------------------|
| | 45 | 15:18:21 | INSERT INTO `simpledb`.`resume` ('userid', 'title', 'skills'...) | 1452: Cannot add or update a child row: a foreign key co... | |
| | 46 | 15:18:58 | SELECT * FROM simpledb.resume LIMIT 0, 1000 | 3 row(s) returned | 0.000 sec / 0.000 sec |
| | 47 | 15:19:27 | SELECT * FROM simpledb.users LIMIT 0, 1000 | 6 row(s) returned | 0.000 sec / 0.000 sec |
| | 48 | 15:19:47 | SELECT * FROM simpledb.resume LIMIT 0, 1000 | 2 row(s) returned | 0.000 sec / 0.000 sec |
| | 49 | 15:21:18 | SELECT * FROM simpledb.users LIMIT 0, 1000 | 5 row(s) returned | 0.000 sec / 0.000 sec |
| | 50 | 15:22:08 | SELECT * FROM simpledb.resume LIMIT 0, 1000 | 2 row(s) returned | 0.000 sec / 0.000 sec |

Context Help: Snippets

По action output видим что изменения касаются сразу обеих таблиц, в таблице resume id пользователя поменялось