

Базы данных

Лабораторная работа 1. Начало работы с MySQL. MySQL Workbench

Отчёт

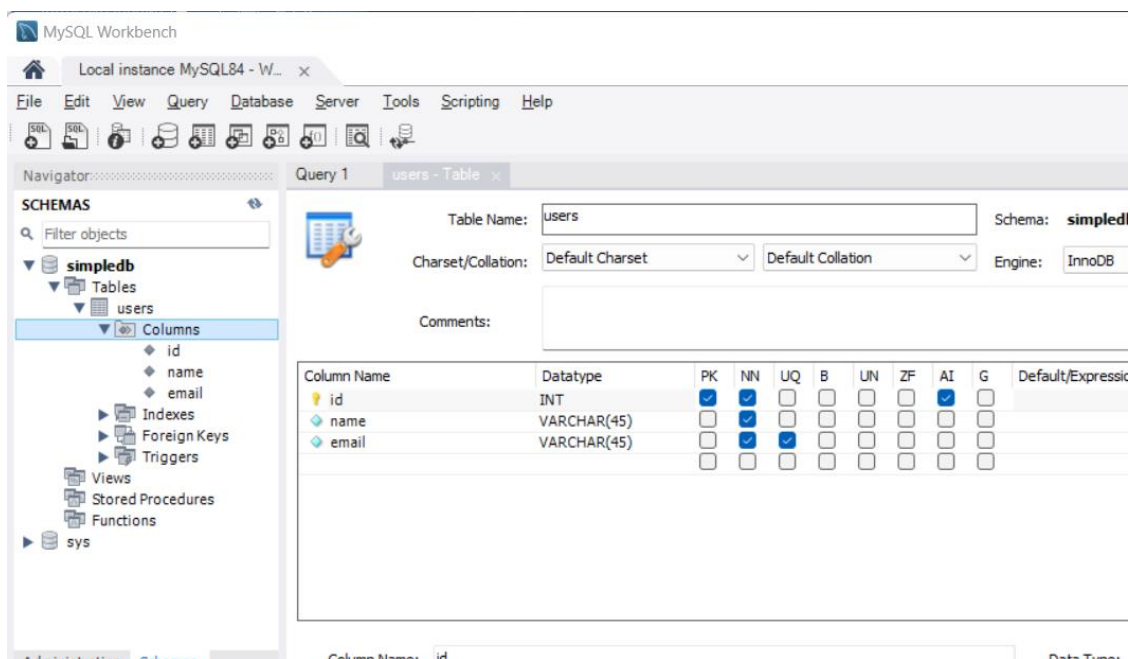
Задание 1 Используя справочные материалы по MySQL Workbench, опишите назначение пунктов меню Management (“Управление”), “Instance” (“Экземпляр БД”) и “Performance” (“Производительность”). В качестве примера описания используйте следующий пример для раздела Management (его описывать не надо).

Описание назначение пунктов меню:

1. Раздел “Instance”:
 - a. Раздел “Startup / Shutdown”.
 - i. В разделе отображается информация о состоянии работы сервера.
 - ii. Сервер можно вручную запустить или остановить.
2. Раздел “Server Logs”.
 - a. Во вкладке “Error Log File” отображается настройка для уровней логирования логов сервера.
 - b. Во вкладке “Show Log File” можно посмотреть содержимое файла.
3. Раздел “Performance”.
 - a. Раздел “Dashboard”.
 - i. В разделе отображается информация эффективности работы:
 1. сети,
 2. mySQL сервера,
 3. пула базы данных.
 - b. Раздел “Performance Report”.
 - i. В разделе отображается информация о отчётах работы базы данных.
 - ii. Можно выделить следующие группы отчётов:
 1. Использование памяти;
 2. Максимальная скорость чтения/записи;
 3. Затратные операции SQL;
 4. Статистика схем базы данных;

5. Время ожидания;
 6. Статистика InnoDB;
 7. Затраты ресурсов пользователя.
4. Раздел “Performance Setup”.
- а. В разделе отображается эффективность работы схемы БД.

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.

Если таблица уже создана, то вы можете щелкнуть правой кнопкой мыши по названию таблицы и выбрать пункт меню Copy to Clipboard - Create Statement.

Команда запроса:

```
CREATE DATABASE `simpledb` /*!40100 DEFAULT CHARACTER SET utf8mb3 */
/*!80016 DEFAULT ENCRYPTION='N' */;
```

Задание 4

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

Команда запроса:

```
INSERT INTO `simplifiedb`.`users` (`id`, `name`, `email`) VALUES ('1', 'Ignat', 'PersonalMailbox@gmail.com');
```

```
INSERT INTO `simplifiedb`.`users` (`id`, `name`, `email`) VALUES ('2', 'Korney', 'korney-chukovsky-inspiration@mail.ru');
```

```
INSERT INTO `simplifiedb`.`users` (`id`, `name`, `email`) VALUES ('3', 'Iter', 'botnetprw2025@gmail.com');
```

Команда обновления элемента:

```
UPDATE `simplifiedb`.`users` SET `name` = 'Alan', `email` = 'TimeToAwake@gmail.com' WHERE (`id` = '1');
```

В таблицу были добавлены: age tinyint(1) – возраст, country tinytext – название страны, country_shortened char(4) – сокращение от названия страны.

Задание 5

Команда запроса:

```
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 DEFAULT CURRENT_TIMESTAMP()  
AFTER `rating`,  
CHANGE COLUMN `name` `name` VARCHAR(50) NOT NULL ,  
;
```

Значение по умолчанию CURRENT_TIMESTAMP() у поля с типом данных – TIMESTAMP означает, что при вставке новой строки в таблицу, если для данного столбца не указано конкретное значение, в него автоматически будет записано текущее системное время.

Пользователь может не хотеть оставить информацию о своей дате рождения и, как следствие, возрасте, стране, поле и почтовом индексе. По этой причине они могут иметь значение по умолчанию NULL. Поля с именем, адресом электронной почты и датой создания необходимы и не могут быть NULL.

Команда обновления значений по умолчанию:

```
ALTER TABLE `simplifiedb`.`users`  
CHANGE COLUMN `name` `name` VARCHAR(50) NOT NULL ,  
CHANGE COLUMN `email` `email` VARCHAR(45) NOT NULL ,  
CHANGE COLUMN `country_shortened` `country_shortened` CHAR(4) NULL DEFAULT  
NULL ;  
CHANGE COLUMN `country` `country` TINYTEXT NULL ,  
CHANGE COLUMN `bday` `bday` DATE NULL ,  
CHANGE COLUMN `created` `created` TIMESTAMP NOT NULL DEFAULT  
CURRENT_TIMESTAMP ;
```

Задание 6

В таблицу были добавлены 4 пользователя вручную и 2 с помощью команд:

id	name	email	age	country_shortened	country	gender	bday	postal_code
1	Alan	TimeToAwake@gmail.com	NULL	Spb	Saint-Petersburg	NULL	NULL	NULL
2	Korney	korney-chukovsky-inspiration@mail.ru	NULL	Mscw	Moscow	NULL	NULL	NULL
3	Iter	botnetprw2025@gmail.com	NULL	Ngct	Nightcity	NULL	NULL	NULL
4	Georgy	longroadtokamchatcka@mail.ru	NULL	Kmct	Kamchatka	NULL	NULL	NULL
5	Ekaterina	ekaterina.petrova@outlook.com	NULL	NULL	NULL	F	2000-02-11	1457
6	Paul	paul@superpochta.ru	NULL	NULL	NULL	M	1998-08-12	1237
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Задание 7

Содержание файла с SQL-запросами:

id,name,email,age,country_shortened,country,gender,bday,postal_code,rating,created

1,Alan,TimeToAwake@gmail.com,NULL,Spb,Saint-Petersburg,NULL,NULL,NULL,NULL,"2025-02-12 21:40:16"

2,Korney,korney-chukovsky-inspiration@mail.ru,NULL,Mscw,Moscow,NULL,NULL,NULL,NULL,"2025-02-12 21:40:16"

3,Iter,botnetprw2025@gmail.com,NULL,Ngct,Nightcity,NULL,NULL,NULL,NULL,"2025-02-12 21:40:16"

4,Georgy,longroadtokamchatcka@mail.ru,NULL,Kmct,Kamchatka,NULL,NULL,NULL,NULL,"2025-02-12 21:40:16"

5,Ekaterina,ekaterina.petrova@outlook.com,NULL,NULL,NULL,F,2000-02-11,145789,1.123,"2025-02-16 16:41:36"

6,Paul,paul@superpochta.ru,NULL,NULL,NULL,M,1998-08-12,123789,1,"2025-02-16 16:42:32"

Информация храниться в виде строк, где первая строка содержит в себе название столбцов, разделённые запятой. Ниже расположены значения для соответствующих названий также разделённые запятой. В местах, где значения не указаны, записано значение по умолчанию – NULL.

Задание 8

Команда создания таблицы:

```
CREATE TABLE `simplified`.`users_info` (
  `resumeid` INT NOT NULL AUTO_INCREMENT,
```

```
`userid` INT NOT NULL,  
`title` VARCHAR(100) NOT NULL,  
`skills` TEXT(1) NOT NULL,  
`created` TIMESTAMP NULL DEFAULT CURRENT_TIMESTAMP,  
PRIMARY KEY (`resumeid`),  
INDEX `userid_idx` (`userid` ASC) VISIBLE,  
CONSTRAINT `userid`  
  FOREIGN KEY (`userid`)  
  REFERENCES `simplified`.`users` (`id`)  
  ON DELETE CASCADE  
  ON UPDATE CASCADE)  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8  
COLLATE = utf8_unicode_ci;
```

Анализ SQL-запроса:

Создается таблица `users_info`, которая содержит следующие поля:

`resumeid` – первичный автоинкрементируемый ключ.

`userid` – внешний ключ, связывающий `users_info` с `users(id)`.

`title` – строка длиной до 100 символов.

`skills` – текстовое поле TEXT.

`created` – временная метка, по умолчанию устанавливаемая в текущее время.

Опция ON DELETE CASCADE указывает, что при удалении записи из `users` автоматически будут удаляться все связанные записи в `users_info`, где `userid` соответствует удаленному `id` из `users`. Это предотвращает наличие записей в `users_info`, которые бы ссылались на несуществующего пользователя.

При удалении записей из users_info, не будут затрагиваться users, так как ограничение внешнего ключа действует только в одном направлении (от users_info к users).

Опция ON UPDATE CASCADE указывает, что при обновлении записи из users автоматически будут обновляться все связанные записи в users_info. При этом обновление users_info не повлечёт обновления users.

Задание 9

Команда добавления резюме:

```
INSERT INTO `simplifiedb`.`users_info` (`userid`, `title`, `skills`) VALUES ('1', 'Full Stack Developer', 'JavaScript, Node.js, SQL');
```

```
INSERT INTO `simplifiedb`.`users_info` (`userid`, `title`, `skills`) VALUES ('2', 'Data Scientist', 'Python, Pandas, Machine Learning, SQL');
```

```
INSERT INTO `simplifiedb`.`users_info` (`userid`, `title`, `skills`) VALUES ('5', 'UI/UX Designer', 'Figma, Adobe XD, Prototyping, User Research');
```

```
INSERT INTO `simplifiedb`.`users_info` (`userid`, `title`, `skills`) VALUES ('6', 'DevOps Engineer', 'Docker, Kubernetes, CI/CD, Cloud Computing');
```

Таблица users_info использует id из users для сопоставления данных резюме с пользователями и при этом каждое из них имеет свой идентификатор. Поэтому у одного пользователя может быть как не быть ни одного резюме, так и неограниченное количество.

Содержание файла:

resumeid,userid,title,skills,created

1,1,"Full Stack Developer","JavaScript, Node.js, SQL","2025-02-13 19:25:45"

2,2,"Data Scientist","Python, Pandas, Machine Learning, SQL","2025-02-13 19:25:45"

3,5,"UI/UX Designer","Figma, Adobe XD, Prototyping, User Research","2025-02-13 19:25:45"

4,6,"DevOps Engineer","Docker, Kubernetes, CI/CD, Cloud Computing","2025-02-13 19:25:45"

Информация храниться в виде строк, где первая строка содержит в себе название столбцов, разделённые запятой. Ниже расположены значения для соответствующих названий также разделённые запятой. Строковые значения с пробелами записаны в кавычках.

При попытке добавить в таблицу resume строчку с userid несуществующего пользователя получаем ошибку:

```
Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails
(`simplifiedb`.`users_info`, CONSTRAINT `userid` FOREIGN KEY (`userid`) REFERENCES
`users` (`id`) ON DELETE CASCADE ON UPDATE CASCADE)
```

Сообщение говорит о невозможности создать элемент userid связанный со значением из таблицы users. Если попытаться добавить запись в таблицу users_info с userid, которого нет в таблице users, произойдет нарушение ссылочной целостности, поскольку userid является внешним ключом (FOREIGN KEY) со ссылкой на users(id). СУБД блокирует такую операцию, чтобы сохранить целостность данных.

Задание 10

При удалении двух пользователей из таблицы users, произойдёт удаление соответствующих записей resume.

Команда удаления пользователей:

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

```
DELETE FROM `simplifiedb`.`users` WHERE (`id` = '5');
```

```
UPDATE `simplifiedb`.`users` SET `id` = '10' WHERE (`id` = '6');
```

До выполнения команды:

Result Grid									
Filter Rows:									
Edit: Export/Import: Wrap Cell Content: IA									
id	name	email	age	country_shortened	country	gender	bday	postal_co	
1	Alan	TimeToAwake@gmail.com	NULL	NULL	NULL	M	NULL	NULL	
2	Korney	korney-chukovsky-inspiration@mail.ru	NULL	Mscw	Moscow	NULL	NULL	NULL	
3	Iter	botnetprw2025@gmail.com	NULL	Ngct	Nightcity	NULL	NULL	NULL	
4	Georgy	longroadtokamchatcka@mail.ru	NULL	Kmct	Kamchatka	NULL	NULL	NULL	
6	Paul	paul@superpochta.ru	NULL	NULL	NULL	M	1998-08-12	123789	
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	

Result Grid									
Filter Rows:									
Edit: Export/Import: Wrap Cell Content: IA									
resumeid	userid	title	skills	created					
1	1	Full S...	JavaScript, Node.js, SQL	2025-02-16 20:13:53					
2	2	Data...	Python, Pandas, Machine Learning, SQL	2025-02-13 19:25:45					
3	5	UI/U...	Figma, Adobe XD, Prototyping, User Research	2025-02-16 20:14:22					
4	6	Dev...	Docker, Kubernetes, CI/CD, Cloud Computing	2025-02-13 19:25:45					
NULL	NULL	NULL	NULL	NULL					

Так как в users_info установлен внешний ключ с ON DELETE CASCADE, удаление пользователя из users приведет к автоматическому удалению всех его связанных записей в users_info.

Так как установлен ON UPDATE CASCADE, если изменить id пользователя в users, то во всех связанных записях в users_info значение userid автоматически обновится.

После выполнения команды

Result Grid									
Filter Rows:									
Edit: Export/Import: Wrap Cell Content: IA									
id	name	email	age	country_shortened	country	gender	bday	postal_co	
2	Korney	korney-chukovsky-inspiration@mail.ru	NULL	Mscw	Moscow	NULL	NULL	NULL	
3	Iter	botnetprw2025@gmail.com	NULL	Ngct	Nightcity	NULL	NULL	NULL	
4	Georgy	longroadtokamchatcka@mail.ru	NULL	Kmct	Kamchatka	NULL	NULL	NULL	
10	Paul	paul@superpochta.ru	NULL	NULL	NULL	M	1998-08-12	123789	
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	

Result Grid									
Filter Rows:									
Edit: Export/Import: Wrap Cell Content: IA									
resumeid	userid	title	skills	created					
2	2	Data...	Python, Pandas, Machine Learning, SQL	2025-02-13 19:25:45					
4	10	Dev...	Docker, Kubernetes, CI/CD, Cloud Computing	2025-02-13 19:25:45					
NULL	NULL	NULL	NULL	NULL					