

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

КАФЕДРА СИСТЕМ ШТУЧНОГО ІНТЕЛЕКТУ



Звіт до лабораторної роботи №3
з дисципліни:
“ОБДЗ”
на тему:
“Модифікація структури таблиць бази даних засобами SQL”

Підготувала:
студентка групи КН-209
Дипко Олександра
Викладач:
Мельникова Н.І.

Львів 2020

Мета роботи:

Розробити SQL запити для зміни структури таблиць бази даних (включення нових полів, вилучення полів таблиць, зміни опису полів, зміни обмежень).

Короткі теоретичні відомості.

Для зміни структури таблиць використовується наступна директива мови SQL.

Синтаксис:

```
ALTER [IGNORE] TABLE ім'я_таблиці  
опис_зміни [, опис_зміни ...]
```

опис_зміни:

```
[додаткові_параметри ...]  
| ADD [COLUMN] назва_поля опис_поля [FIRST  
| AFTER назва_поля ]  
| ADD [COLUMN] (назва_поля опис_поля, ...)  
| ADD {INDEX|KEY} [ім'я_обмеження] (назва_поля, ...) [тип_обмеження] |  
ADD [CONSTRAINT [ім'я_обмеження]] PRIMARY KEY (назва_поля, ...)  
[тип_обмеження]  
| ADD [CONSTRAINT [ім'я_обмеження]]  
UNIQUE [INDEX|KEY] [ім'я_обмеження] (назва_поля, ...) [тип_обмеження] | ADD  
[FULLTEXT|SPATIAL] [INDEX|KEY] [ім'я_обмеження] (назва_поля, ...)  
[тип_обмеження]  
| ADD [CONSTRAINT [ім'я_обмеження]]  
FOREIGN KEY [ім'я_обмеження] (назва_поля, ...) опис_зв'язку  
| ALTER [COLUMN] назва_поля {SET DEFAULT значення | DROP DEFAULT} |  
CHANGE [COLUMN] стара_назва_поля нова_назва_поля опис_поля  
[FIRST|AFTER назва_поля]  
| MODIFY [COLUMN] назва_поля опис_поля  
[FIRST | AFTER назва_поля]  
| DROP [COLUMN] назва_поля  
| DROP PRIMARY KEY  
| DROP {INDEX|KEY} ім'я_обмеження  
| DROP FOREIGN KEY ім'я_обмеження  
| RENAME [TO] нова_назва_таблиці  
| ORDER BY назва_поля [, назва_поля] ...  
| CONVERT TO CHARACTER SET кодування [COLLATE набір_правил]  
| [DEFAULT] CHARACTER SET [=] кодування [COLLATE [=] набір_правил]
```

додаткові_параметри:

```
{ENGINE|TYPE} [=] тип_таблиці  
| AUTO_INCREMENT [=] початкове_значення |  
| CHECKSUM [=] {0 | 1}  
| [DEFAULT] COLLATE [=] набір_правил  
| COMMENT [=] 'коментар до таблиці' |  
DELAY_KEY_WRITE [=] {0 | 1}  
| MAX_ROWS [=] значення  
| MIN_ROWS [=] значення  
| ROW_FORMAT {DEFAULT|DYNAMIC|FIXED|COMPRESSED|REDUNDANT|COMPACT}  
AVG_ROW_LENGTH [=] значення  
| [DEFAULT] CHARACTER SET [=] кодування
```

Аргументи:

тип_обмеження

Вказує тип індексу для поля: USING {BTREE | HASH | RTREE}.

IGNORE

Ігнорувати помилки дублювання значень в ключових полях, тобто залишати в таблиці перше значення, і не вставляти дублікат. За замовчуванням, при такій помилці вся дія над таблицею відміняється.

ALTER *назва_поля*

Використовується для зміни значення за замовчуванням для поля.

MODIFY *назва_поля*

Змінити тип поля.

CHANGE *назва_поля*

Змінити назву і тип поля.

FIRST | AFTER

Змінити розташування поля в таблиці. За замовчуванням, нове поле додається в кінець таблиці. FIRST додасть його на початок; AFTER – одразу після вказаного поля.

DROP *назва_поля*

Видалення поля, обмеження чи індексу.

ORDER BY

Відсортувати рядки в таблиці за зростанням значень у вказаному полі (полях).

AUTO_INCREMENT

При зміні початкового значення для приросту поля-лічильника, слід вказати значення більше за ті, які вже містяться в цьому полі.

CONVERT TO

Задає кодування, в яке потрібно перевести таблицю.

RENAME TO

Змінити назву таблиці на вказану. Для зміни назви таблиці, або для переносу таблиці в іншу базу, також можна використати директиву RENAME TABLE ім'я_бази_1.назва_таблиці_1 TO ім'я_бази_2.назва_таблиці_2.

Хід роботи.

Виконаємо команди: SHOW DATABASES, USE LIBRARY, SHOW TABLES, DESCRIBE BOOK.

```
mysql> use library;
Database changed
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| library          |
| mysql            |
| performance_schema |
| sakila           |
| sys              |
| world            |
+-----+
7 rows in set (0.00 sec)

mysql> show tables;
+-----+
| Tables_in_library |
+-----+
| author            |
| book              |
| book_author       |
| country           |
| genre             |
| genre_book        |
| publish_house     |
| reader            |
| reader_book       |
+-----+
9 rows in set (0.00 sec)
```

```
mysql> describe book;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_book | int | NO | PRI | NULL | auto_increment |
| name | char(45) | NO | | NULL | |
| year | year | YES | | NULL | |
| page | int | YES | | NULL | |
| id_publish_house | int | NO | MUL | NULL | |
| description_of_book | varchar(1000) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

Виконаємо такі операції над схемою бази даних.

1. Вилучимо зв'язок між книжкою та видавництвом.
2. Вилучимо і змінимо поля в таблиці **book**.
3. Змінимо поле в таблиці **author**.
4. Додамо поле і нове обмеження унікальності до таблиці **publish_house**.
5. Змінимо тип обмеження цілісності для зв'язку `file_message`.

1. Видалення зв'язку між таблицями шляхом видалення обмеження для таблиці:

До:

```
| book | CREATE TABLE `book` (  
  `id_book` int NOT NULL AUTO_INCREMENT,  
  `name` char(45) NOT NULL,  
  `year` year DEFAULT NULL,  
  `page` int DEFAULT NULL,  
  `id_publish_house` int NOT NULL,  
  `description_of_book` varchar(1000) DEFAULT NULL,  
  PRIMARY KEY (`id_book`),  
  KEY `id_publish_house_idx` (`id_publish_house`),  
  CONSTRAINT `id_publish_house` FOREIGN KEY (`id_publish_house`) REFERENCES `publish_house` (`id_publish_house`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

ALTER TABLE book

```
DROP FOREIGN KEY id_publish_house,  
DROP INDEX id_publish_house_idx;
```

```
mysql> ALTER TABLE book  
-> DROP FOREIGN KEY id_publish_house, DROP INDEX id_publish_house_idx;  
Query OK, 0 rows affected (0.26 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

Для перевірки результату можна виконати команди `SHOW INDEX FROM book` та `SHOW CREATE TABLE book`.

Після:

```
| book | CREATE TABLE `book` (  
  `id_book` int NOT NULL AUTO_INCREMENT,  
  `name` char(45) NOT NULL,  
  `year` year DEFAULT NULL,  
  `page` int DEFAULT NULL,  
  `id_publish_house` int NOT NULL,  
  `description_of_book` varchar(1000) DEFAULT NULL,  
  PRIMARY KEY (`id_book`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

2. Видалення поля і зміна розмірності поля в таблиці File:

До:

Field	Type	Null	Key	Default	Extra
id_reader	int	NO	PRI	NULL	auto_increment
name	varchar(45)	NO		NULL	
surname	varchar(45)	NO		NULL	
age	varchar(45)	NO		NULL	
email	varchar(45)	NO		NULL	
password	varchar(45)	NO		NULL	

ALTER TABLE reader

```
DROP COLUMN age,
```

```
MODIFY COLUMN surname varchar(22) NULL;
```

```
mysql> ALTER TABLE reader  
-> DROP COLUMN age,  
-> MODIFY COLUMN surname varchar(22) NULL;  
Query OK, 0 rows affected (1.50 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

Після:

```
mysql> describe reader;
```

Field	Type	Null	Key	Default	Extra
id_reader	int	NO	PRI	NULL	auto_increment
name	varchar(45)	NO		NULL	
surname	varchar(22)	YES		NULL	
email	varchar(45)	NO		NULL	
password	varchar(45)	NO		NULL	

3. Зробимо поле Name в таблиці Author не обов'язковим, і змінимо його розмірність:

До:

Field	Type	Null	Key	Default	Extra
id_author	int	NO	PRI	NULL	auto_increment
firstname	varchar(45)	NO		NULL	
surname	varchar(45)	YES		NULL	
nickname	varchar(45)	YES		NULL	
born_date	date	NO		NULL	
death_date	date	YES		NULL	
id_country	int	NO	MUL	NULL	

ALTER TABLE author

MODIFY COLUMN name CHAR(60) NULL;

```
mysql> ALTER TABLE author
-> MODIFY COLUMN firstname CHAR(60) NULL;
Query OK, 0 rows affected (1.54 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

Після:

```
mysql> describe author;
```

Field	Type	Null	Key	Default	Extra
id_author	int	NO	PRI	NULL	auto_increment
firstname	char(60)	YES		NULL	
surname	varchar(45)	YES		NULL	
nickname	varchar(45)	YES		NULL	
born_date	date	NO		NULL	
death_date	date	YES		NULL	
id_country	int	NO	MUL	NULL	

4. Додамо поле і нове обмеження унікальності до таблиці publish house.

Додамо до таблиці publish_house поле id_sample, яке дозволить зберігати різні версії одного і того ж повідомлення. Спочатку змінимо поле id_publish_house, забравши лічильник (властивість auto_increment). Після додавання поля id_sample, створимо нове обмеження унікальності для полів (id_publish_house, id_sample).

До:

Field	Type	Null	Key	Default	Extra
id_publish_house	int	NO	PRI	NULL	auto_increment
name	varchar(45)	NO		NULL	
address	varchar(45)	YES		NULL	
contact_person	varchar(45)	YES		NULL	

ALTER TABLE publish_house

MODIFY id_publish_house INT(22) UNSIGNED NOT NULL,

ADD COLUMN id_sample TINYINT(8) UNSIGNED NOT NULL **AFTER** id_publish_house,

ADD CONSTRAINT UNIQUE UQ_publish_house_sample (id publish house, id sample);

```
mysql> ALTER TABLE publish_house
-> MODIFY id_publish_house INT(22) UNSIGNED NOT NULL,
-> ADD COLUMN id_sample TINYINT(8) UNSIGNED NOT NULL AFTER id_publish_house,
-> ADD CONSTRAINT UNIQUE UQ_publish_house_sample (id_publish_house, id_sample);
Query OK, 0 rows affected, 2 warnings (1.54 sec)
Records: 0 Duplicates: 0 Warnings: 2
```

Після:

```
mysql> describe publish_house;
```

Field	Type	Null	Key	Default	Extra
id_publish_house	int unsigned	NO	PRI	NULL	
id_sample	tinyint unsigned	NO		NULL	
name	varchar(45)	NO		NULL	
address	varchar(45)	YES		NULL	
contact_person	varchar(45)	YES		NULL	

5. Змінимо тип обмеження цілісності для зв'язку book_author.

Змінимо обмеження цілісності book_author так, щоб при видаленні книги, з бази видалялись записи про всі приєднані до нього файли. Для цього потрібно видалити обмеження і створити нове, оскільки MySQL не підтримує директиви ALTER CONSTRAINT.

До:

```
| book_author | CREATE TABLE `book_author` (  
  `id` int NOT NULL AUTO_INCREMENT,  
  `id_book_author` int NOT NULL,  
  `id_author` int NOT NULL,  
  PRIMARY KEY (`id`),  
  KEY `id_author_idx` (`id_author`),  
  KEY `id_book_author_idx` (`id_book_author`),  
  CONSTRAINT `id_author` FOREIGN KEY (`id_author`) REFERENCES `author` (`id_author`),  
  CONSTRAINT `id_book_author` FOREIGN KEY (`id_book_author`) REFERENCES `book` (`id_book`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

ALTER TABLE book_author

DROP FOREIGN KEY id_book_author,

ADD CONSTRAINT id_book_author **FOREIGN KEY** (id_book_author)
REFERENCES library.book (id_book) **ON DELETE** CASCADE;

Після (SHOW CREATE TABLE book_author):

```
| book_author | CREATE TABLE `book_author` (  
  `id` int NOT NULL AUTO_INCREMENT,  
  `id_book_author` int NOT NULL,  
  `id_author` int NOT NULL,  
  PRIMARY KEY (`id`),  
  KEY `id_author_idx` (`id_author`),  
  KEY `id_book_author_idx` (`id_book_author`),  
  CONSTRAINT `id_author` FOREIGN KEY (`id_author`) REFERENCES `author` (`id_author`),  
  CONSTRAINT `id_book_author` FOREIGN KEY (`id_book_author`) REFERENCES `book` (`id_book`) ON DELETE CASCADE  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Висновок:

на цій лабораторній роботі було розглянуто методи модифікації структури таблиць БД засобами SQL та проведено модифікацію полів, зв'язків і обмежень.

