

Лабораторна робота №3

з курсу “ОБДЗ”

на тему:

“Модифікація структури таблиць бази даних засобами SQL”

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  
| AVG_ROW_LENGTH [=] значення  
| [DEFAULT] CHARACTER SET [=] кодування
```

| 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}

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 TABLES; DESCRIBE FILE.

```
mysql> show tables;
```

Tables_in_mycms
author
category
comment
file
message
message_category
role
session

8 rows in set (0.01 sec)

```
mysql> describe file;
```

Field	Type	Null	Key	Default	Extra
fileID	int(10) unsigned	NO	PRI	NULL	auto_increment
messageID	int(10) unsigned	NO	MUL		
authorID	int(10) unsigned	NO	MUL		
fname	char(39)	NO			
path	char(199)	NO			
type	char(5)	NO			
size	int(10) unsigned	YES		NULL	

7 rows in set (0.00 sec)

```
mysql> _
```

Виконаємо такі операції над схемою бази даних.

1. Вилучимо зв'язок file_author.
2. Вилучимо і змінимо поля в таблиці File.
3. Змінимо поле в таблиці Author.
4. Додамо поле і нове обмеження унікальності до таблиці Message.
5. Змінимо тип обмеження цілісності для зв'язку file_message.

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

```
ALTER TABLE MyCMS.FILE
    DROP FOREIGN KEY file_author,
    DROP INDEX file_author;
```

Для перевірки результату можна виконати команди SHOW INDEX FROM file, та SHOW CREATE TABLE file.

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

```
ALTER TABLE MyCMS.FILE
    DROP COLUMN authorID,
    MODIFY COLUMN size int(22) NULL;
```

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

```
ALTER TABLE MyCMS.AUTHOR
    MODIFY COLUMN name CHAR(60) NULL;
```

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

```
ALTER TABLE MyCMS.MESSAGE
    MODIFY messageID INT(32) UNSIGNED NOT NULL,
    ADD COLUMN versionID TINYINT(8) UNSIGNED NOT NULL AFTER
messageID,
    ADD CONSTRAINT UNIQUE UQ_Message_version (messageID,
versionID);
```

Виконаємо команду DESCRIBE MESSAGE.

```
mysql> describe message;
```

Field	Type	Null	Key	Default	Extra
messageID	int(32) unsigned	NO	PRI		
versionID	tinyint(8)	NO			
message	text	NO			
authorID	int(22) unsigned	NO	MUL		
posted	datetime	NO			
status	enum('published','unpublished','deleted')	YES		published	
rating	tinyint(4)	YES		NULL	
attachment	tinyint(1)	YES		0	

```
8 rows in set (0.02 sec)
```

5. Змінимо обмеження цілісності file_message так, щоб при видаленні повідомлення, 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.

```
ALTER TABLE MyCMS.FILE
    DROP FOREIGN KEY file_message,
    ADD CONSTRAINT file_message FOREIGN KEY (messageID)
REFERENCES MyCMS.MESSAGE (messageID) ON DELETE CASCADE;
```

Для перевірки результату можна виконати команду SHOW CREATE TABLE file.

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