

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»**

Інститут комп'ютерних наук та інформаційних технологій

Кафедра систем штучного інтелекту



Звіт до лабораторної роботи №4

з дисципліни

“Організація Баз Даних та знань”

Виконала:

ст. гр. КН-210
Заремба Вікторія

Викладач:

Мельникова Н.І.

Лабораторна робота №4
з курсу “ОБДЗ”
на тему:
“Запити на додавання, зміну та вилучення даних”

Мета роботи: Розробити SQL-запити для внесення нових значень в таблиці в режимі одиничного та групового доповнення; розробити SQL-запити для внесення змін в рядки таблиць; розробити SQL-запити для вилучення вибраних рядків.

Короткі теоретичні відомості.

Для внесення значень в таблиці використовується директива INSERT.

```
INSERT [LOW_PRIORITY | DELAYED | HIGH_PRIORITY] [IGNORE]
[INTO] назва_таблиці [(назва_поля,...)]
[{VALUES | VALUE} ({вираз | DEFAULT},...),(...),...
| [SET назва_поля={вираз | DEFAULT}, ...]
| [вибірка_даних]
[ON DUPLICATE KEY UPDATE назва_поля=вираз [, назва_поля=вираз]...]
```

Аргументи:

VALUES

Задає набір значень, які будуть вставлятися у задані поля таблиці з дотриманням вказаного у дужках порядку полів. Якщо перелік полів не задано, то значення потрібно задати для кожного поля таблиці.

DEFAULT

Внесення значення за замовчуванням у вказане поле.

вираз

Константа, змінна або інший вираз (див. далі). Наприклад: поле2=поле1*0.5.

вибірка_даних

Довільна SELECT-директива, результати виконання якої будуть внесені в таблицю.

ON DUPLICATE KEY UPDATE

Якщо при внесенні нового рядка, значення в ключових полях дублюються, то ця директива дозволяє задати вираз, за яким будуть формуватись нові значення полів наявного в таблиці (старого) рядка. При цьому, новий рядок не буде внесений в таблицю.

Наступна директива MySQL дозволяє заносити дані в таблицю із зовнішнього файлу.

```
LOAD DATA [LOW_PRIORITY | CONCURRENT] [LOCAL] INFILE 'назва_файлу'
[REPLACE | IGNORE]
INTO TABLE назва_таблиці
[CHARACTER SET кодування]
[{FIELDS | COLUMNS}
[TERMINATED BY 'рядок_завершення']
[[OPTIONALLY] ENCLOSED BY 'символ']
[ESCAPED BY 'символ']]
[LINES [STARTING BY 'рядок_початку']
[TERMINATED BY 'рядок_завершення']]
[IGNORE кількість LINES]
```

Для внесення змін в записи таблиці використовується директива UPDATE.

UPDATE [LOW_PRIORITY] [IGNORE] назва_таблиці
SET назва_поля1={вираз|DEFAULT} [, назва_поля2={вираз|DEFAULT}]...
[WHERE умова_відбору]
[ORDER BY ...]
[LIMIT кількість_рядків]

Аргументи:

SET

Задає список полів, значення яких будуть змінюватись, і відповідні дії над ними.

DEFAULT

Вставка значення по замовчуванню у вказане поле.

WHERE

Вказує умову, за якою відбираються рядки, що підлягають зміні.

ORDER BY

Може вказувати поле, за яким сортувати рядки перед зміною, з метою уникнення порушення цілісності таблиці.

LIMIT

Обмежує кількість рядків, які будуть змінені.

Для видалення записів з таблиці використовується директива DELETE.

DELETE [LOW_PRIORITY] [QUICK] [IGNORE] FROM назва_таблиці
[WHERE умова_відбору]
[ORDER BY ...]
[LIMIT кількість_рядків]

Аргументи:

FROM

Вказує таблицю, з якої слід видалити значення.

WHERE

Вказує умову, за якою відбираються рядки, що підлягають видаленню.

ORDER BY

Може вказувати поле, за яким сортувати рядки перед видаленням.

LIMIT

Обмежує кількість рядків, які будуть видалені.

При побудові виразів та запитів, можна використовувати оператори та функції MySQL.
Опишемо основні з них у таблиці.

Функція (оператор)	Опис
AND, OR	Логічне “і” та логічне “або”.
~, &, , ^	Побітові операції інверсії, логічного “і”, “або”, “виключного або”.
AVG(), MAX(), MIN()	Повертає середнє, максимальне, або мінімальне значення для аргументів.

ABS()	Повертає модуль числа.
POW(<i>X</i> , <i>Y</i>)	Повертає число <i>X</i> у степіні <i>Y</i> .
RAND()	Повертає псевдовипадкове число з плаваючою крапкою.
NOT IN()	Перевіряє, чи не входить значення до множини вказаних значень.
NULLIF(<i>вираз1</i> , <i>вираз2</i>)	Повертає NULL, якщо значення рівні між собою.
IF(<i>вираз1</i> , <i>вираз2</i> , <i>вираз3</i>)	Повертає <i>вираз2</i> , якщо виконується <i>вираз1</i> . В протилежному випадку повертає <i>вираз3</i> .
SIN(); COS(); TAN(); COT(); LN()	Повертає значення синуса, косинуса, тангенса, котангенса, натурального логарифма.
BETWEEN ... AND ...	Перевіряє, чи входить значення у заданий діапазон.
COUNT()	Рахує кількість рядків, які повертає запит.
COUNT(DISTINCT)	Рахує кількість різних значень.
CHAR_LENGTH(<i>рядок</i>)	Повертає кількість символів в аргументі.
CONCAT(<i>рядок1</i> , ...)	Повертає об'єднані рядки.
NOW(), CURRENT_TIMESTAMP()	Повертає поточну дату і час у форматі 'YYYY-MM-DD HH:MM:SS'.
CURRENT_DATE(), CURRENT_DATE	Повертає поточну дату.
CURRENT_TIME(), CURRENT_TIME	Повертає поточний час.
DAYOFMONTH(<i>дата</i>)	Повертає день місяця з отриманої дати.
MONTH(<i>дата</i>)	Повертає номер місяця, присутнього у отриманій даті.
YEAR(<i>дата</i>)	Повертає рік з отриманої дати.
STRCMP(<i>рядок1</i> , <i>рядок2</i>)	Порівнює два рядки.
LIKE <i>рядок</i>	Порівняння з шаблоном. Можна використовувати % (довільні символи) і _ (довільний символ).
RLIKE <i>шаблон</i>	Порівняння з шаблоном, яке підтримує регулярні вирази.
MATCH (<i>поля</i>) AGAINST (<i>рядок</i>)	Здійснює пошук рядка у вказаних символьних полях таблиці. (Тільки для MyISAM-таблиць.)
ROW_COUNT()	Рахує кількість рядків, оновлених останнім запитом.
BENCHMARK(<i>к-сть</i> , <i>вираз</i>)	Виконує вираз вказану кількість разів і повертає час виконання. Замість виразу можна вказати запит, який повинен повертати максимум одне значення.
COMPRESS(<i>рядок_символів</i>), UNCOMPRESS(<i>рядок_символів</i>)	Якщо MySQL встановлений з бібліотекою <i>zlib</i> , функції дозволяють стискати і розпаковувати символьні рядки.
AES_ENCRYPT(<i>значення</i> , <i>ключ</i>), AES_DECRYPT(<i>значення</i> , <i>ключ</i>)	128-бітне шифрування і дешифрування рядків за заданим ключем.
MD5(<i>значення</i>)	Обчислює MD5-хеш заданого рядка.

Хід роботи

Виконаємо такі запити для занесення даних у таблиці і їх подальшої модифікації.

1. Заповнимо таблицю User в режимі одиночного і групового доповнення.
2. Створимо файли з даними і заповнимо ними решту таблиць баз даних.
3. Виконаємо модифікацію значень у таблиці budget.
4. Видалимо записи з таблиці task.

1. Заповнимо таблицю User в режимі одиночного і групового доповнення.

Внесення нових значень в таблицю Role в режимі одиночного доповнення:

```
INSERT INTO user VALUES (1, 'Zaremba', 'Viktoria', 'vika241020@gmail.com', '+380985796738', '12345678');
```

Внесення нових значень в таблицю Role в режимі групового доповнення:

```
INSERT INTO user
VALUES (NULL, 'Vyshnevska', 'Anastasia', 'urbica@gmail.com', '+123456789012', 'jlkdc'),
(NULL, 'Boyechko', 'Denys', 'Denys@gmail.com', '+158956789012', 'vdsvzc'),
(NULL, 'Popova', 'Daryna', 'daryna@gmail.com', '+123908789012', 'f8uejiosdak'),
(NULL, 'Kondratyk', 'Bogdana', 'bogdana@gmail.com', '+789456789012', 'cdjvx'),
(NULL, 'Rogynski', 'Roman', 'roman@gmail.com', '+124568789012', 'jocs,lm'),
(NULL, 'Osovska', 'Sophia', 'sonya@gmail.com', '+789458909012', 'uhdosj.k');
```

Таблиця user після виконання запиту:

	id_user	surname	name	mail_address	phone_number	password
1	1	Zaremba	Viktoria	vika241020@gmail.com	+380985796738	12345678
2	2	Vyshnevska	Anastasia	urbica@gmail.com	+123456789012	jlkdc
3	3	Boyechko	Denys	Denys@gmail.com	+158956789012	vdsvzc
4	4	Popova	Daryna	daryna@gmail.com	+123908789012	f8uejiosdak
5	5	Kondratyk	Bogdana	bogdana@gmail.com	+789456789012	cdjvx
6	6	Rogynski	Roman	roman@gmail.com	+124568789012	jocs,lm
7	7	Osovska	Sophia	sonya@gmail.com	+789458909012	uhdosj.k

2. Створимо файли з даними і заповнимо ними решту таблиць баз даних.

Приклад завантаження даних з файлу:

```
LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\task.csv'
INTO TABLE task
CHARACTER SET CP1251
FIELDS TERMINATED BY ';'
LINES TERMINATED BY '\n'
(name,deadline,id_department)
SET id_task=NULL;
```

Для того, щоб в базі даних коректно відображались усі символи використовуємо кодування CP1251

Для автогенерування PR задаємо `SET id_task=NULL`, а перед цим вказуємо в дужках, які поля потрібно заповнити з файлу. Не вказані обов'язкові поля прийматимуть дефолтне значення, необов'язкові поля можуть залишатись пустими (якщо не було вказане дефолтнезначення)

3. Виконаємо модифікацію значень у таблиці project.

Змінюємо назви проектів за ID. Для зручності відображення даних сортуємо за датою.

```
UPDATE project SET name = "Курінний табір"
WHERE id_project=2;
UPDATE project SET name = "Вишкыл проектного менеджменту"
WHERE id_project=3;
SELECT * FROM project
ORDER BY date;
```

Таблиця до зміни значень:

	name	date	purpose	p...	tar...	comm...
1	1 By St. Nicolas steps	2020-12-18	0		Nicolaiv	activ people
2	2 PM trainig	2020-02-20	0	develop PM skills	Molodvizh...	scouts
3	3 Summer Camp	2020-07-03	1	teambealding		scouts

Таблиця після зміни значень і сортування

	name	date	purpose	place	target_audience
1	2 Курінний табір	2020-02-20	0 develop PM skills	Molodvizh centre	scouts
2	3 Вишкыл проектного менеджменту	2020-07-03	1 teambealding		scouts
3	1 By St. Nicolas steps	2020-12-18	0	Nicolaiv	activ people

4. Видалимо записи з таблиці user.

```
DELETE FROM user  
WHERE surname = 'Popova';
```

Таблиця до видалення інформації:

	id_user	surname	name	mail_address	phone_number	password
1	1	Zarembo	Viktoria	vika241020@gmail.com	+380985796738	12345678
2	2	Vyshnevskaya	Anastasia	urbica@gmail.com	+123456789012	jlkd
3	3	Boyechko	Denys	Denys@gmail.com	+158956789012	vdsvzc
4	4	Popova	Daryna	daryna@gmail.com	+123908789012	f8uejiosdak
5	5	Kondratyuk	Bogdana	bogdana@gmail.com	+789456789012	cdjvx
6	6	Rogynski	Roman	roman@gmail.com	+124568789012	jocs,lm
7	7	Osovskaya	Sophia	sonya@gmail.com	+789458909012	uhdosj.k
8	8	Мацьків	Настя	nastya@gmail.com	+789450009012	ujdsdosj.k
9	9	Корінь	Оля	olya@gmail.com	+789458909012	udjsksj.k
10	10	Осташевська	Надія	nadia@gmail.com	+789458909012	uhvdjkj.k
11	11	Драбик	Ярина	yaryna@gmail.com	+789458909012	uhdosj.kjdj
12	12	Зубик	Vika	vika@gmail.com	+789458909012	uhdokdjv.k

Таблиця після видалення користувача:

	id_user	surname	name	mail_address	phone_number	password
1	1	Zarembo	Viktoria	vika241020@gmail.com	+380985796738	12345678
2	2	Vyshnevskaya	Anastasia	urbica@gmail.com	+123456789012	jlkd
3	3	Boyechko	Denys	Denys@gmail.com	+158956789012	vdsvzc
4	5	Kondratyuk	Bogdana	bogdana@gmail.com	+789456789012	cdjvx
5	6	Rogynski	Roman	roman@gmail.com	+124568789012	jocs,lm
6	7	Osovskaya	Sophia	sonya@gmail.com	+789458909012	uhdosj.k
7	8	Мацьків	Настя	nastya@gmail.com	+789450009012	ujdsdosj.k
8	9	Корінь	Оля	olya@gmail.com	+789458909012	udjsksj.k
9	10	Осташевська	Надія	nadia@gmail.com	+789458909012	uhvdjkj.k
10	11	Драбик	Ярина	yaryna@gmail.com	+789458909012	uhdosj.kjdj
11	12	Зубик	Vika	vika@gmail.com	+789458909012	uhdokdjv.k

Висновок: у цій лабораторній роботі було розглянуто способи наповнення і модифікації даних в таблицях БД та проведено модифікацію даних у двох таблицях.