МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

КАФЕДРА СИСТЕМ ШТУЧНОГО ІНТЕЛЕКТУ



Звіт до лабораторної роботи №4

з дисципліни:

“ОБДЗ”

на тему:

“ **Запити на додавання, зміну та вилучення даних**”

**Підготувала:**

студентка групи КН-209

Дипко Олександра

**Викладач:**

Мельникова Н.І.

Львів 2020

**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* [, н*азва\_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.

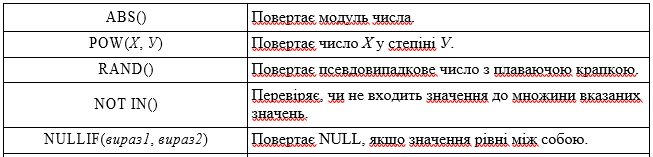
LIMIT

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. |
| ROW\_COUNT() | Рахує кількість рядків, оновлених останнім запитом. |
| COMPRESS(*рядок\_символів*), | Якщо MySQL встановлений з бібліотекою *zlib*, функції |
| UNCOMPRESS(*рядок\_символів*) | дозволяють стискати і розпаковувати символьні рядки. |
|  |  |
| AES\_ENCRYPT(*значення*, *ключ*), | 128-бітне шифрування і дешифрування рядків за |
| AES\_DECRYPT(*значення*, *ключ*) | заданим ключем. |
| MD5(*значення*) | Обчислює MD5-хеш заданого рядка. |



|  |  |
| --- | --- |
|  |  |
| IF(*вираз1*, *вираз2*, *вираз3*) | Повертає *вираз2*, якщо виконується *вираз1*. В |
| протилежному випадку повертає *вираз3*. |
|  |
|  |  |
| 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(*рядок*) | Повертає кількість символів в аргументі. |
|  |  |
| CONCAT(*рядок1*, ...) | Повертає об’єднані рядки. |
|  |  |
| NOW(), | Повертає поточну дату і час у форматі |
| СURRENT\_TIMESTAMP() | 'YYYY-MM-DD HH:MM:SS'. |
|  |  |
| CURRENT\_DATE(), | Повертає поточну дату. |
| CURRENT\_DATE |
|  |
|  |  |
| CURRENT\_TIME(), | Повертає поточний час. |
| CURRENT\_TIME |
|  |
|  |  |
| DAYOFMONTH(*дата*) | Повертає день місяця з отриманої дати. |
|  |  |
| MONTH(*дата*) | Повертає номер місяця, присутнього у отриманій даті. |
|  |  |
| YEAR(*дата*) | Повертає рік з отриманої дати. |
|  |  |
| STRCMP(*рядок1*, *рядок2*) | Порівнює два рядки. |
|  |  |
| LIKE *рядок* | Порівняння з шаблоном. Можна використовувати % |
| (довільні символи) і \_ (довільний символ). |
|  |
|  |  |
| RLIKE шаблон | Порівняння з шаблоном, яке підримує регулярні вирази. |
|  |  |
| MATCH (*поля*) AGAINST (*рядок*) | Здійснює пошук рядка у вказаних символьних полях |
| таблиці. (Тільки для MyISAM-таблиць.) |
|  |  |

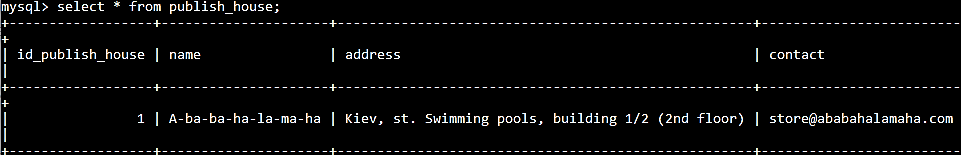
**Хід роботи.**

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

1. Заповнимо таблицю **publish\_house** в режимі одиночного і групового доповнення.
2. Створимо файл з даними і заповнимо таблицю **author**.
3. Заповнимо решту таблиць бази даних.
4. Виконаємо модифікацію значень у таблиці **reader**.
5. Видалимо записи з таблиці **reader**.
6. Заповнимо таблицю **publish\_house** в режимі одиночного і групового доповнення.

* Внесення нових значень в таблицю publish\_house в режимі одиничного доповнення:

**INSERT INTO** publish\_house(id\_publish\_house,name,address,contact) **VALUES** (1,'A-ba-ba-ha-la-ma-ha','Kiev, st. Swimming pools, building 1/2 (2nd floor)','store@ababahalamaha.com');



* Внесення нових значень в таблицю publish\_house в режимі групового доповнення:

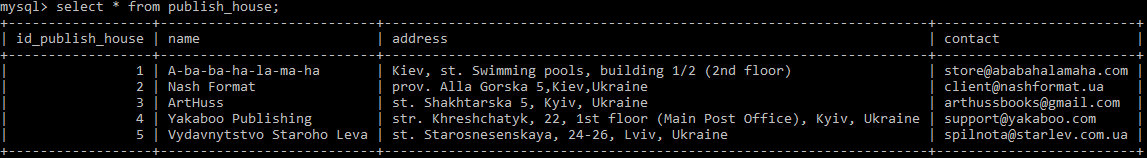
**INSERT INTO** publish\_house (name,address,contact)

**VALUES** ('Nash Format','prov. Alla Gorska 5,Kiev,Ukraine','client@nashformat.ua'),

('ArtHuss','st. Shakhtarska 5, Kyiv, Ukraine','arthussbooks@gmail.com'),

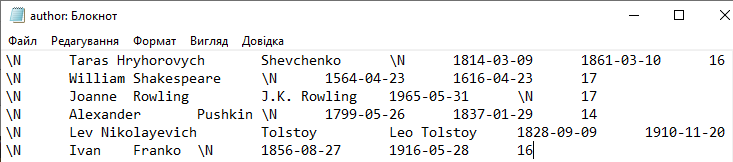
('Yakaboo Publishing','str. Khreshchatyk, 22, 1st floor (Main Post Office), Kyiv, Ukraine','support@yakaboo.com'),

('Vydavnytstvo Staroho Leva','st. Starosnesenskaya, 24- 26, Lviv, Ukraine','spilnota@starlev.com.ua');

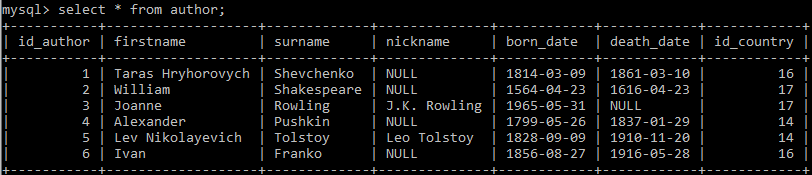


1. Створимо файл з даними і заповнимо таблицю **author.**

Значення полів повинні розділятись символом табуляції, а кожен рядок таблиці повинен починатися з нового рядка у файлі. Виконаємо дві наступні команди.



**LOAD DATA INFILE** 'author.txt' **INTO TABLE** author;



1. Заповнимо решту таблиць бази даних.

INSERT INTO country (country)

VALUES ('Armenia'),

("Australia"),

("Austria"),

('Belgium'),

("Britain"),

('Canada'),

('China'),

('Cuba'),

('France'),

("Germany"),

('Ireland'),

('Italy'),

('Poland'),

('Russia'),

('Sweden'),

('Ukraine'),

('United Kingdom'),

('United States');



INSERT INTO genre (name, about)

VALUES ("Fantasy", "A fantasy story is about magic or supernatural forces, happens to take place in a modern or future era."),

("Adventure", "An adventure story is about a protagonist who journeys to epic or distant places to accomplish something."),

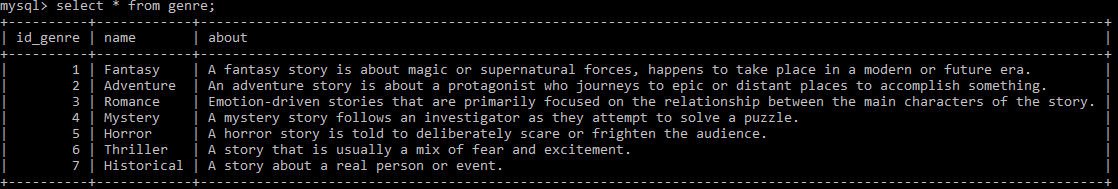
("Romance", "Emotion-driven stories that are primarily focused on the relationship between the main characters of the story."),

("Mystery", "A mystery story follows an investigator as they attempt to solve a puzzle."),

("Horror", "A horror story is told to deliberately scare or frighten the audience."),

("Thriller", "A story that is usually a mix of fear and excitement."),

("Historical", "A story about a real person or event.");



INSERT INTO reader (name,surname,age,email,password)

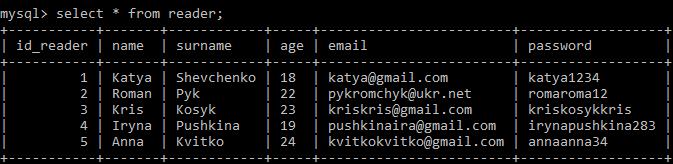
VALUES ('Katya','Shevchenko',18,'katya@gmail.com','katya1234'),

('Roman','Pyk',22,'pykromchyk@ukr.net','romaroma12'),

('Kris','Kosyk',23,'kriskris@gmail.com','kriskosykkris'),

('Iryna','Pushkina',19,'pushkinaira@gmail.com','irynapushkina283'),

('Anna','Kvitko',24,'kvitkokvitko@gmail.com','annaanna34');



INSERT INTO book (name,year,page,id\_publish\_house,description\_of\_book)

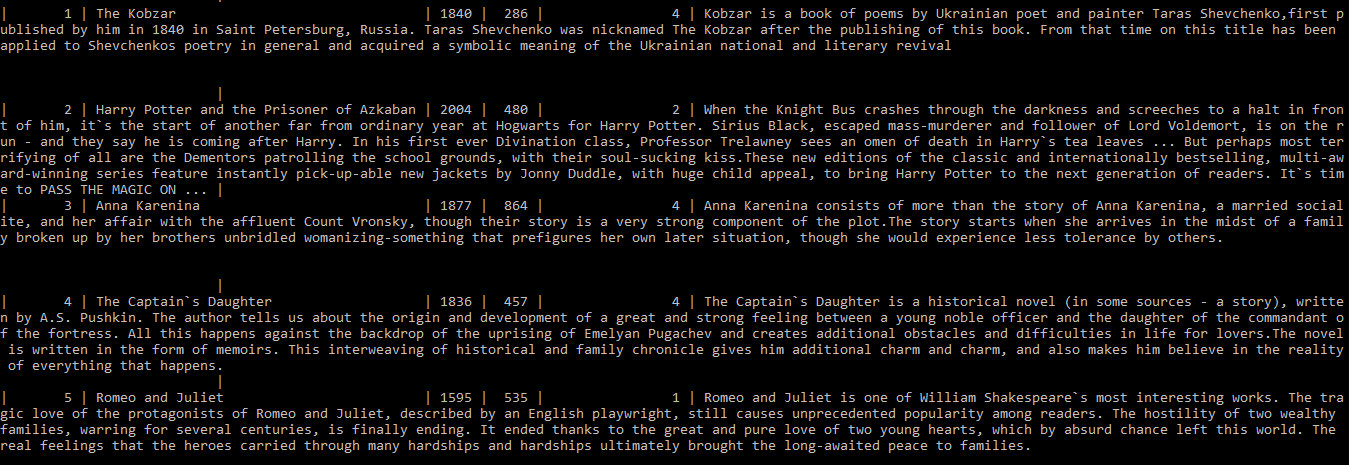
VALUES ('The Kobzar',1840,'286',4,'…'),

('Harry Potter and the Prisoner of Azkaban',2004,'480',2,'...'),

('Anna Karenina',1877,'864',4, '…'),

('The Captain`s Daughter',1836,'457',4,'…'),

('Romeo and Juliet',1595,'535',1,'…');



INSERT INTO book\_author(id\_book\_author, id\_author)

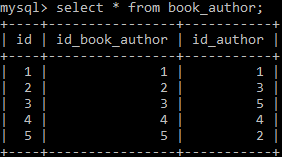
VALUES (1,1),

(2,3),

(3,5),

(4,4),

(5,2);



INSERT INTO genre\_book(id\_book, id\_genre)

VALUES (2,1),

(2,2),

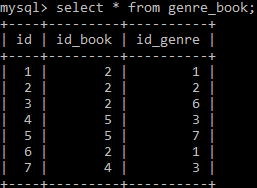
(2,6),

(5,3),

(5,7),

(2,1),

(4,3);



INSERT INTO reader\_book (id\_reader\_book, id\_reader, upload\_date)

VALUES (1,2, '2020-02-11'),

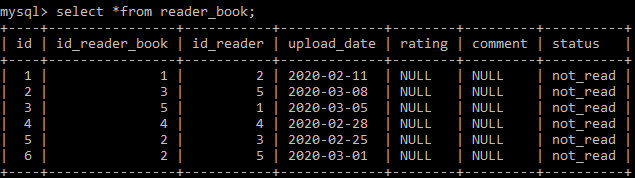
(3,5, '2020-03-08'),

(5,1, '2020-03-05'),

(4,4, '2020-02-28'),

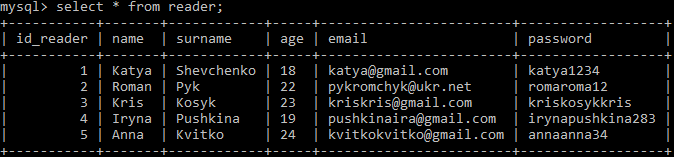
(2,3, '2020-02-25'),

(2,5, '2020-03-01');



1. Виконаємо модифікацію значень у таблиці **reader**.

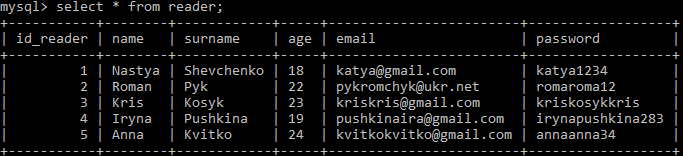
**До:**



**UPDATE** reader **SET** name = ‘Nastya’

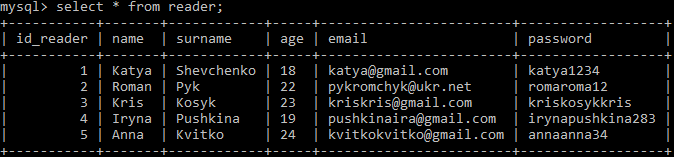
**WHERE** id\_reader = 1;

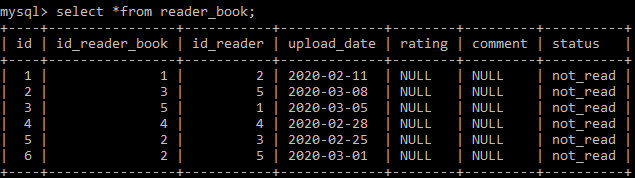
**Після:**



1. Видалимо записи з таблиці **reader.**

**До:**

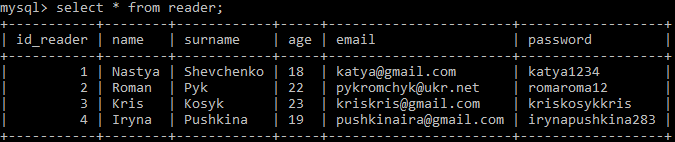


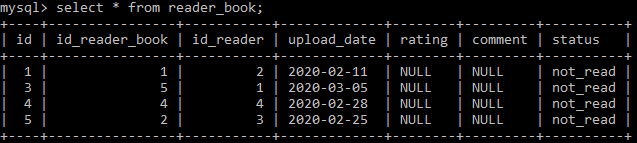


**DELETE FROM** reader **WHERE** id\_reader = 5;

Також видаляться дані з **reader\_book**, оскільки ми використовували CASCADE.

**Після:**





**Висновок:**

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