

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

КАФЕДРА СИСТЕМ ШТУЧНОГО ІНТЕЛЕКТУ



Звіт до лабораторної роботи №7
з дисципліни:
“ОБДЗ”
на тему:
“Запити на вибір даних з таблиць бази даних”

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

Львів 2020

Мета роботи:

Розробити SQL запити відбору даних з одиничних та з'єднаних таблиць, в тому числі з використанням підзапитів, натурального, умовного та лівого з'єднання, із застосуванням у критеріях вибірки функцій та операторів, в т. ч. LIKE, BETWEEN, IS NULL, IS NOT NULL, IN (...), NOT IN (...), ALL, SOME, ANY, EXISTS.

Короткі теоретичні відомості

Для вибирання даних з таблиць використовується директива SELECT, яка може містити інші директиви SELECT (підзапити, або вкладені запити) та директиви з'єднання таблиць.

SELECT

```
[ALL | DISTINCT | DISTINCTROW ] [STRAIGHT_JOIN]
[SQL_CACHE | SQL_NO_CACHE] [SQL_CALC_FOUND_ROWS]
елемент_вибірки [, елемент_вибірки ...] [FROM перелік_таблиць]
[WHERE умова_відбору]
[GROUP BY {ім'я_поля | синонім | позиція_поля} [ASC | DESC], ...]
[HAVING умова_відбору]
[ORDER BY {ім'я_поля | синонім | позиція_поля} [ASC | DESC], ...]
[LIMIT {к-сть_рядків [OFFSET зміщення]} [PROCEDURE ім'я_процедури(аргументи) ]
[INTO OUTFILE 'ім'я_файлу' опції_експорту
| INTO DUMPFILE 'ім'я_файлу'
| INTO змінна [, змінна]]
```

Параметри:

SELECT

Вказує поля, константи та вирази, що будуть відображатися у результатах запиту. Директива вимагає чіткого дотримання порядку ключових слів FROM, WHERE і т.д.

елемент_вибірки

Вказує елемент, який буде включатися в результати відбору. Такими елементами можуть бути: ім'я поля, константа або вираз. Кожному елементу можна присвоїти ім'я-псевдонім, яке буде відображатись у результатах запиту. Для цього після назви елемента слід дописати AS псевдонім.

перелік_таблиць

Назви таблиць, з яких здійснюється вибір значень. Тут можна задавати синоніми назвам таблиць (ім'я_таблиці AS синонім), використовувати підзапити SELECT для формування таблиці з вказаним синонімом, з'єднувати декілька таблиць.

WHERE

Вказує критерії порівняння (або підзапити) для відбору рядків.

Явним чином вмикає/вимикає зберігання результатів запиту у кеші запитів MySQL. За замовчуванням, кешування запитів залежить від системної змінної query_cache_type.

SQL_CALC_FOUND_ROWS

Вказує, що при виконанні запиту слід обчислити загальну кількість рядків в результаті, ігноруючи опцію обмеження LIMIT. Цю кількість рядків потім можна отримати командою SELECT FOUND_ROWS().

Для вибору записів зі з'єднаних таблиць використовується директива SELECT разом із директивами JOIN у переліку таблиць. Наприклад:

```
SELECT * FROM author INNER JOIN comment
ON author.authorID = comment.authorID;
```

INTO

Вказує місце, куди будуть збережені результати запиту. Це може бути як зовнішній файл, так і параметри чи змінні, визначені користувачем. Кількість змінних має бути рівна кількості полів у результаті.

DISTINCT | DISTINCTROW

Видалення з результату рядків-дублікатів. За замовчуванням вибираються всі рядки.

STRAIGHT_JOIN

Опція, яка строго задає порядок вибирання кортежів зі з'єднуваних таблиць в порядку переліку таблиць. (Оптимізатор запитів MySQL іноді змінює цей порядок.)

SQL_CACHE | SQL_NO_CACHE

У таблиці нижче описано основні функції порівняння, які можна використовувати при формуванні складних критеріїв вибору.

Функція	Опис
STRCMP(<i>рядок1, рядок2</i>)	Порівнює два рядки. Повертає значення 0 (False) якщо рядки однакові, -1 якщо перший рядок менший за другий, і 1 (True) в усіх інших випадках.
LIKE <i>рядок</i>	Порівняння з рядком-шаблоном. В шаблоні можна використовувати знаки % (довільні символи) і _ (довільний символ).
REGEXP <i>рядок</i>	Порівняння з рядком з використанням регулярних виразів. Функція-синонім – RLIKE.
MATCH (<i>поля</i>) AGAINST (<i>рядок</i>)	Здійснює пошук рядка у вказаних текстових полях таблиці. (Тільки для MyISAM-таблиць.)
BETWEEN ... AND ...	Повертає 1, якщо значення належить даному діапазону.
NOT BETWEEN ... AND ...	Повертає 1, якщо значення не належить діапазону.
IN(<i>арг1, арг2, ...</i>)	Перевірка належності множині. Повертає 1, якщо значення співпадає хоча б із одним аргументом, і 0 – у протилежному випадку. Повертає NULL, якщо значення є NULL, або якщо співпадіння не знайдено, а один із аргументів є NULL.
NOT IN(<i>арг1, арг2, ...</i>)	Повертає 1, якщо значення не міститься у множині аргументів, і 0 – у протилежному випадку. Повертає NULL аналогічно до функції IN().
IS NULL, IS NOT NULL	Перевірка визначеності значення.
LEAST(<i>арг1, арг2, ...</i>)	Повертає мінімальне значення серед аргументів. Повертає NULL, якщо хоча б один із аргументів є NULL.
GREATEST(<i>арг1, арг2, ...</i>)	Повертає максимальне значення серед аргументів. Повертає NULL, якщо хоча б один із аргументів є NULL.

Для формування критеріїв вибору та підзапитів також використовують наступні оператори порівняння:

=

Оператор перевірки рівності двох виразів. Якщо відбувається порівняння двох не NULL значень, то повертає значення 1 (True) коли обидва вирази рівні, інакше результатом є значення 0 (False). Якщо хоча б один з виразів приймає значення NULL, то результатом є значення NULL.

<=>

Перевірка рівності виразів, яке враховує NULL значення. Повертає 1, якщо обидва вирази приймають значення NULL, або рівні значення. Повертає 0, якщо один із виразів приймає значення NULL, або значення виразів не рівні.

>, >=

Порівняння двох виразів. Результатом є 1, якщо ліве значення більше (більше рівне) ніж праве, інакше результатом є 0. Якщо хоча б один з виразів приймає значення NULL, то результатом теж стає NULL.

<, <=

Порівняння двох виразів. Результатом є 1, якщо ліве значення менше (менше рівне) ніж праве, інакше результатом є 0. Якщо хоча б один з виразів приймає значення NULL, то результатом теж є NULL.

!=, <>

Перевірка на не рівність. Результат набуває значення 1, якщо ліве значення менше або більше ніж праве, інакше результатом є 0. Якщо хоча б один з виразів приймає значення NULL, то результатом теж є NULL.

ALL, SOME, ANY

Оператори, які можна використовувати після операторів порівняння. Задають необхідність виконання оператора хоча б для одного (SOME, ANY) чи всіх (ALL) елементів, отриманих в результаті підзапиту. На відміну від функцій IN (), NOT IN () оператори не працюють зі списками значень.

[NOT] EXISTS

Оператор, який використовують після ключового слова WHERE. Повертає 1, якщо підзапит повертає хоча б одне визначене значення, і 0 – у протилежному випадку.

Хід роботи

1. Показати пароль заданого користувача.
2. Показати користувачів і їхні коментарі до книги (ліве з'єднання таблиць).
3. Показати перелік книг за жанром Romance (натуральне з'єднання), спочатку виконавши проєкцію genre_book2.
4. Показати всі коментарі до книг за жанром Romance та Historical (умовне з'єднання).
5. Показати останні 3 коментарі до книг за жанром Romance та Historical (підзапит).
6. Визначити користувачів, які не написали жодного повідомлення.
7. Визначити користувачів, паролі яких не відповідають вимогам безпеки (менші за 8 символів або не містять цифр).

1. Показати пароль заданого користувача.

```
SELECT password
FROM reader WHERE id_reader = 3;
```

```
mysql> select password
-> FROM reader WHERE id_reader = 3;
+-----+
| password |
+-----+
| kriskosykkris |
+-----+
```

2. Показати користувачів і їхні коментарі до книги (ліве з'єднання таблиць).

```
SELECT reader.id_reader, reader.name, reader.email, reader_book.comment,
reader_book.rating
FROM reader LEFT JOIN reader_book ON
reader.id_reader = reader_book.id_reader;
```

id_reader	name	email	comment	rating
1	Nastya	katya@gmail.com	NULL	NULL
2	Roman	pykromchyk@ukr.net	I will recommend it to my friends!	10
3	Kris	kriskris@gmail.com	Woow, interesting book!	9
4	Iryna	pushkinaira@gmail.com	Amazing start!	NULL

3. Показати перелік книг за жанром Romance (натуральне з'єднання), спочатку виконавши проєкцію genre_book2.

```
CREATE VIEW genre_book2 AS SELECT DISTINCT genre_book.id_book,
genre_book.id_genre, book.name
FROM genre_book, book WHERE genre_book.id_book = book.id_book;
```

id_book	id_genre	name
2	1	Harry Potter and the Prisoner of Azkaban
2	2	Harry Potter and the Prisoner of Azkaban
2	6	Harry Potter and the Prisoner of Azkaban
5	3	Romeo and Juliet
5	7	Romeo and Juliet
4	3	The Captain's Daughter

```
SELECT genre_book2.name, genre.name
FROM genre_book2 INNER JOIN genre ON genre.id_genre =
genre_book2.id_genre
WHERE genre.name = "Romance";
```

name	name
Romeo and Juliet	Romance
The Captain's Daughter	Romance

4. Показати всі коментарі до книг за жанром Romance та Historical (умовне з'єднання).

```
SELECT genre_book2.name, genre.name, reader_book.comment, reader_book.rating
FROM ( genre_book2 INNER JOIN genre ) INNER JOIN reader_book
ON genre.id_genre = genre_book2.id_genre
AND reader_book.id_reader_book = genre_book2.id_book
WHERE genre.name IN ("Romance", "Historical");
```

name	name	comment	rating
Romeo and Juliet	Romance	NULL	8
Romeo and Juliet	Historical	NULL	8
The Captain's Daughter	Romance	Amazing start!	10
Anna Karenina	Historical	NULL	NULL

5. Показати останні 3 коментарі до книг за жанром Romance та Historical (підзапит).

```
SELECT genre_book2.name, genre.name, reader_book.comment, reader_book.rating
FROM ( genre_book2 INNER JOIN genre ) INNER JOIN reader_book
ON genre.id_genre = genre_book2.id_genre
AND reader_book.id_reader_book = genre_book2.id_book
WHERE genre.name IN ("Romance", "Historical") ORDER BY reader_book.rating DESC LIMIT 3;
```

name	name	comment	rating
Romeo and Juliet	Romance	NULL	8
Romeo and Juliet	Historical	NULL	8
Anna Karenina	Historical	NULL	7

6. Визначити користувачів, які не написали жодного повідомлення.

```
SELECT id_reader, name
FROM reader
WHERE id_reader NOT IN (SELECT id_reader FROM reader_book WHERE status IN
("want to read", "in_progress", "read"));
```

id_reader	name
1	Nastya

7. Визначити користувачів, паролі яких не відповідають вимогам безпеки (менші за 8 символів або не містять цифр).

Таблиця reader:

id_reader	name	surname	age	email	password
1	Nastya	Shevchenko	18	katya@gmail.com	katya1234
2	Roman	Pyk	22	pykromchuk@ukr.net	romaroma12
3	Kris	Kosyk	23	kriskris@gmail.com	kriskosykkris
4	Iryna	Pushkina	19	pushkinaira@gmail.com	irynapushkina283

```
SELECT email, password AS pass
FROM reader
WHERE CHAR_LENGTH(password)<8 OR
(password)NOT REGEXP '[0-9]';
```

email	pass
kriskris@gmail.com	kriskosykkris

Висновок:

на цій лабораторній роботі було вивчено методи вибору даних зі з'єднаних таблиць БД засобами SQL та виконано запити до бази даних з використанням директив SELECT та JOIN, а також складних критеріїв в умові вибірки.