

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

КАФЕДРА СИСТЕМ ШТУЧНОГО ІНТЕЛЕКТУ



Звіт до лабораторної роботи №9
з дисципліни:
“ОБДЗ”
на тему:
«Аналітичні та підсумкові запити»

Виконала:

студентка групи КН-211

Досяк Ірина

Викладач:

Якимишин Х.Я.

Лабораторна робота №9

Мета роботи: розробити SQL запити для вибору записів з однієї чи кількох таблиць із застосуванням агрегатних функцій для отримання підсумкових значень полів.

Короткі теоретичні відомості

Для побудови аналітичних та підсумкових запитів на SQL використовують директиву GROUP BY, а також агрегатні функції:

1. MAX(), MIN() - знаходить максимальне, або мінімальне значення для заданих аргументів.
2. AVG() - знаходить середнє значення для заданих аргументів.
3. AVG(DISTINCT ...) - знаходить середнє значення не враховуючи повтори.
4. SUM() - обчислює суму значень.
5. SUM(DISTINCT ...) - обчислює суму значень.
6. COUNT() - рахує кількість рядків, які повертає запит.
7. COUNT(DISTINCT ...) - рахує кількість різних значень.

Для застосування агрегатних функцій SUM або AVG з часовими типами даних потрібно проводити двосторонню конвертацію типів за допомогою спеціальних функцій, наведених нижче.

TO_DAYS() – перевести дату у число, що означає кількість днів.

FROM_DAYS() – перевести кількість днів у дату.

TIME_TO_SEC() – перевести значення часу у секунди.

SEC_TO_TIME() – перевести кількість секунд у час.

Завдання:

1. Визначити кількість книг у кожній публікації.
2. Вивести кількість нових користувачів за кожен місяць та рік.
3. Визначити середню довжину коментарів для кожного користувача.
4. Визначити найактивнішого в попередньому місяці читача.

Виконання:

1. Визначити кількість книг у кожній публікації.

```
SELECT publish_house.name AS publish_house, COUNT(book.name) AS number
FROM book INNER JOIN publish_house
ON book.id_publish_house=publish_house.id_publish_house
GROUP BY publish_house.name;
```

Результати запиту:

| | publish_house | number |
|---|---------------------|--------|
| ▶ | Yakaboo Publishing | 3 |
| | Nash Format | 1 |
| | A-ba-ba-ha-la-ma-ha | 1 |

1. Вивести кількість нових користувачів за кожен місяць та рік.

```
SELECT YEAR(upload_date) AS year, MONTHNAME(upload_date) AS month,
COUNT(id_reader) AS users
FROM reader_book
GROUP BY year, month WITH ROLLUP HAVING GROUPING(year) = 0;
```

Результати запитів:

| | year | month | users |
|---|------|----------|-------|
| ▶ | 2019 | March | 1 |
| | 2019 | NULL | 1 |
| | 2020 | February | 4 |
| | 2020 | June | 1 |
| | 2020 | March | 1 |
| | 2020 | NULL | 6 |

2. Визначити середню довжину повідомлень для кожного користувача.

```
SELECT name AS reader, reader_book.comment,
AVG(CHAR_LENGTH(reader_book.comment)) AS lenght
FROM reader INNER JOIN reader_book ON reader.id_reader=reader_book.id_reader
GROUP BY reader;
```

Результати запиту:

| | reader | comment | lenght |
|---|--------|------------------------------------|---------|
| ▶ | Katya | I am passionate about this book! | 24.5000 |
| | Roman | Woow, interesting book! | 15.5000 |
| | Kris | Amazing start! | 14.0000 |
| | Iryna | I do not like this! | 19.0000 |
| | Anna | I will recommend it to my friends! | 34.0000 |

3. Визначити найактивнішого за останній місяць читача

Рейтинг активності буде визначатись за к-стю коментарів. Умова відбору WHERE буде відбирати лише учасників, які вже прочитали книгу і тільки ті коментарі, які були написані за попередній місяць (MONTH(CURRENT_DATE)-1).

```
SELECT reader.name AS best_reader, COUNT(DISTINCT reader_book.comment) AS rating
FROM reader INNER JOIN reader_book ON reader.id_reader =
reader_book.id_reader
WHERE MONTH(reader_book.upload_date)=(MONTH(CURRENT_DATE)-1) AND
reader_book.status = "read"
GROUP BY best_reader ORDER BY rating DESC LIMIT 1;
```

Результат запиту:

| | best_reader | rating |
|---|-------------|--------|
| ▶ | Roman | 2 |

Висновок: на цій лабораторній роботі було розглянуто методи застосування агрегатних функцій до результатів вибору даних з таблиць БД

