Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 4 з дисципліни

«Бази даних»

«Створення запитів на групування, сортування, використання вбудованих функцій. Створення та керування представленнями»

Варіант 15

Виконала студентка ІП-12 Кушнір Ганна Вікторівна

Перевірила Марченко Олена Іванівна

Київ 2022

**Лабораторна робота №4**

**Мета:**

* Вивчити оператор, котрий використовується в реляційних СУБД, для вибірки даних з таблиць, групування та сортування даних
* Навчитись використовувати вбудовані функції в запитах
* Вивчити призначення представлень (view) баз даних, синтаксису та семантики команд SQL для їх створення, зміни та видалення, системних збережених процедур для отримання інформації про представлення

**Теоретичні основи**

Викладені в лекційному матеріалі

1. **Постановка задачі лабораторної роботи № 4**

При виконанні комп’ютерного практикуму необхідно виконати наступні дії:

1. Створити наступні запити:
2. запит з використанням функції COUNT;
3. запит з використанням функції SUM;
4. запит з використанням функцій UPPER, LOWER;
5. запит з використанням функцій для роботи з датами;
6. запит з використанням групування по декільком стовпцям;
7. запит з використанням умови відбору груп HAVING;
8. запит з використанням HAVING без GROUP BY;
9. запит з використанням функцій row\_number() over ….;
10. запит з використанням сортування по декільком стовпцям.
11. Робота з представленнями (view):
12. створити представлення, котре містить дані з декількох таблиць;
13. створити представлення, котре містить дані з декількох таблиць та використовує представлення, котре створене в п.a;
14. модифікувати представлення з використанням команди ALTER VIEW;
15. отримати довідникову інформацію про ці представлення з використанням вбудованих процедур (наприклад в MsSQL sp\_help, sp\_helptext та sp\_depends).
16. **Завдання варіанту**

***Програмне забезпечення готелю.***

Основна задача програмного забезпечення, котре проєктується - є відстеження фінансової сторони роботи готелю. Діяльність організована наступним чином: готель надає номери клієнтам на певний термін. Кожен номер характеризується місткістю, комфортністю (люкс, напівлюкс, звичайний, тощо) і ціною. Клієнтами є різні особи, по яким збирається певна інформація (прізвище, ім'я, по-батькові, номер та серія паспорту або іншого документу, що посвідчує особу, додаткова інформація). Здача номера клієнтові проводиться за наявності вільних місць в номерах, за параметрами, котрі вказав клієнт. При поселенні фіксується дата поселення, при виїзді – дата звільнення. номеру. Необхідно не лише зберігати інформацію за фактом здачі номера клієнтові, але і здійснювати бронювання номерів. Крім того, для постійних клієнтів, а також для певних категорій клієнтів передбачена система знижок. Знижки можуть підсумовуватися.

1. **Виконання**
2. Створення запитів з використанням вбудованих функцій, групування та сортування

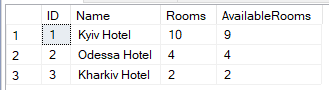
-- a. запит з використанням функції COUNT --

SELECT ID, Name,

(SELECT COUNT(\*) FROM Rooms WHERE Rooms.HotelID = Hotels.ID) AS Rooms,

(SELECT COUNT(\*) FROM Rooms WHERE Rooms.HotelID = Hotels.ID AND Rooms.IsAvailable = 1) AS AvailableRooms

FROM Hotels;



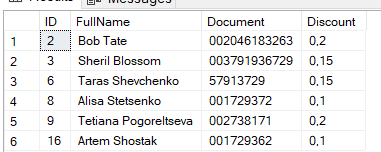
-- b. запит з використанням функції SUM --

SELECT ID, Name + ' ' + Surname AS FullName, Document,

(SELECT ROUND(SUM(Categories.Discount), 3) FROM ClientCategory, Categories WHERE ClientCategory.ClientID = Clients.ID AND Categories.ID = ClientCategory.CategoryID) AS Discount

FROM Clients

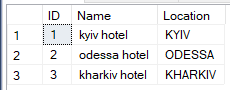
WHERE EXISTS (SELECT \* FROM ClientCategory, Categories WHERE ClientCategory.ClientID = Clients.ID AND Categories.ID = ClientCategory.CategoryID)



-- c. запит з використанням функцій UPPER, LOWER --

SELECT Hotels.ID, LOWER(Hotels.Name) AS Name, UPPER(Hotels.Location) AS Location

FROM Hotels;

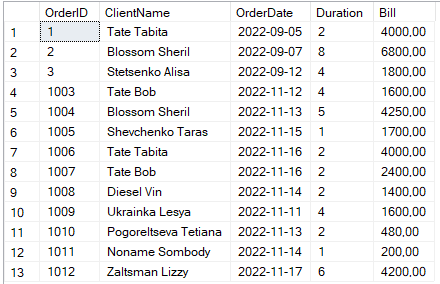


-- d. запит з використанням функцій для роботи з датами --

SELECT Orders.ID AS OrderID, Clients.Surname + ' ' + Clients.Name AS ClientName, Orders.OrderDate, DATEDIFF(day, Orders.CheckInDate, Orders.CheckOutDate) AS Duration, Orders.Bill

FROM Orders, Clients

WHERE Orders.ClientID = Clients.ID;

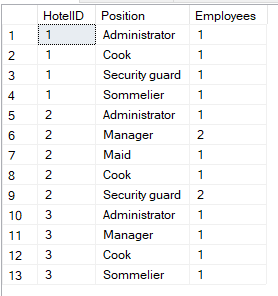


-- e. запит з використанням групування по декільком стовпцям --

SELECT HotelID, (SELECT Name FROM Positions WHERE Positions.ID = Employees.PositionID) AS Position, COUNT(\*) AS Employees

FROM Employees

GROUP BY PositionID, HotelID;



-- f. запит з використанням умови відбору груп HAVING --

SELECT HotelID, COUNT(\*) AS Employees

FROM Employees

GROUP BY HotelID

HAVING COUNT(\*) > 4;



-- g. запит з використанням HAVING без GROUP BY --

SELECT SUM(Amount) AS FullPayroll

FROM Payroll

HAVING SUM(Amount) > 200000;



-- h. запит з використанням функцій row\_number() over ... --

SELECT ROW\_NUMBER() OVER(ORDER BY Surname ASC, Name ASC) AS Row, Surname + ' ' + Name AS FullName, Document

FROM Clients;



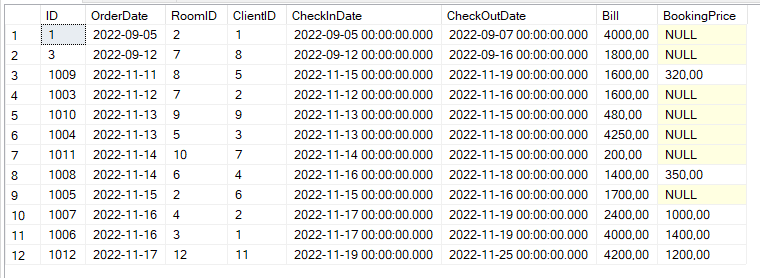
-- i. запит з використанням сортування по декільком стовпцям --

SELECT \*

FROM Orders

WHERE Bill < 5000

ORDER BY OrderDate ASC, CheckInDate ASC, CheckOutDate ASC, Bill ASC;



1. Створення представлень

-- a. створити представлення, котре містить дані з декількох таблиць --

CREATE VIEW Employees\_KyivHotel

AS SELECT T1.ID, T1.Surname, T1.Name, T1.Patronymic, T1.Document, T2.Name Position

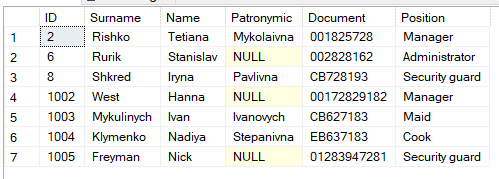
FROM Employees AS T1, Positions AS T2

WHERE T1.PositionID = T2.ID

AND T1.HotelID = 2;

SELECT \*

FROM Employees\_KyivHotel;



-- b. створити представлення, котре містить дані з декількох таблиць та використовує представлення, котре створене в п.a --

CREATE VIEW Employees\_KyivHotel\_Salary

AS SELECT T1.ID, T1.Surname, T1.Position, T2.Salary

FROM Employees\_KyivHotel AS T1, Positions AS T2

WHERE T1.Position = T2.Name;

SELECT \*

FROM Employees\_KyivHotel\_Salary;



-- c. модифікувати представлення з використанням команди ALTER VIEW --

ALTER VIEW Employees\_KyivHotel\_Salary

AS SELECT T1.ID, T1.Surname, T1.Position, T2.Salary

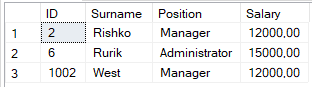
FROM Employees\_KyivHotel AS T1, Positions AS T2

WHERE T1.Position = T2.Name

AND Salary > 7000;

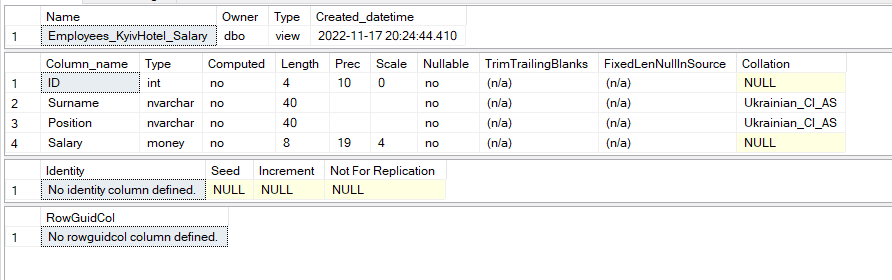
SELECT \*

FROM Employees\_KyivHotel\_Salary;

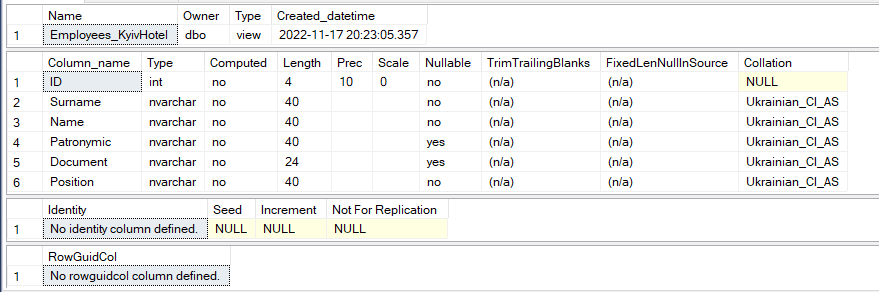


-- d. отримати довідникову інформацію про ці представлення з використанням вбудованих процедур –

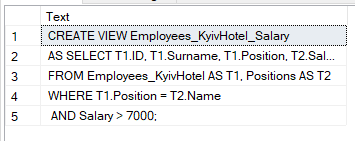
EXEC sp\_help @objname = N'Employees\_KyivHotel\_Salary';



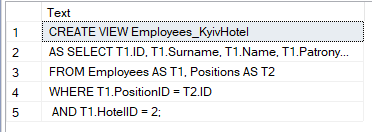
EXEC sp\_help @objname = N'Employees\_KyivHotel';



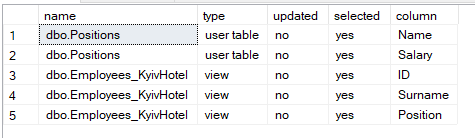
EXEC sp\_helptext @objname = N'Employees\_KyivHotel\_Salary';



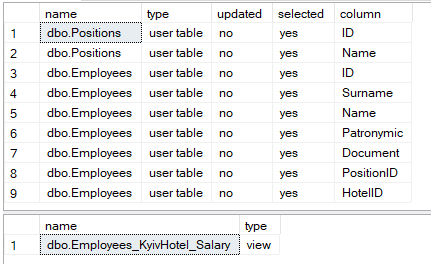
EXEC sp\_helptext @objname = N'Employees\_KyivHotel';



EXEC sp\_depends @objname = N'Employees\_KyivHotel\_Salary';



EXEC sp\_depends @objname = N'Employees\_KyivHotel';



1. **Висновок**

Під час виконання лабораторної роботи з теми «Створення запитів на групування, сортування, використання вбудованих функцій. Створення та керування представленнями» було отримано навички використання вбудованих функцій в SQL-запитах та створення запитів на групування та сортування. Було вивчено оператор, який використовується в реляційних СУБД для вибірки даних з таблиць, групування та сортування даних. Було досліджено призначення представлень (view) баз даних, вивчено синтаксис та семантику команд SQL для їх створення, зміни та видалення, системних збережених процедур для отримання інформації про представлення.