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 №3 з дисципліни

«Бази даних»

# «Побудова простих запитів»

Варіант 3

Виконав студент ІП-13 Бондаренко Максим Вікторович

(шифр, прізвище, ім'я, по батькові)

Перевірив Марченко Олена Іванівна

(прізвище, ім'я, по батькові)

Київ 2022

**Лабораторна робота №3**

**Мета:**

* Вивчити оператор, котрий використовується в реляційних СУБД, для вибірки даних з таблиць.
* Вивчити команди SQL для створення запитів з використанням підзапитів та з’єднань.

**Постановка задачі**

При виконанні комп‘ютерного практикуму необхідно виконати наступні дії:

1) Створити запити для вибірки даних з використанням (разом 12 запитів):

a. Найпростіших умов

b. Операторів порівняння

c. Умов з використанням логічних операторів AND, OR та NOT.

d. Умов з використанням комбінацій логічних операторів

e. З використанням виразів над стовпцями, як в якості новостворених

стовпців, так і умовах

f. Використання операторів:

i. Приналежності множині

ii. Приналежності діапазону

iii. Відповідності шаблону

iv. Перевірка на невизначене значення

2) Створити запити з використанням підзапитів та з’єднань (разом 11

запитів) (в запитах повинні використовуватись 2 та більше таблиць):

a. Використання підзапитів в рядку вибірки полів (у секції select) та

вибірки з таблиць (у секції from)

b. Використання підзапитів в умовах з конструкціями EXISTS, IN

c. Декартовий добуток

d. З’єднання декількох таблиць за рівністю

e. З’єднання декількох таблиць за рівністю та умовою відбору

f. Внутрішнього з’єднання

g. Лівого зовнішнього з’єднання

h. Правого зовнішнього з’єднання

i. Об’єднання запитів

**Виконання завдання**

**Запити**

*use* db\_labs;  
  
*select* \*  
*from* employee;  
  
*select* \*  
*from* contract  
*where* salary > 3500;  
  
*select* \*  
*from* contract  
*where* start\_date = '2022-01-01' *or* salary >= 4000;  
  
*select* \*  
*from* contract  
*where* start\_date = '2022-01-01' *and* salary >= 4000;  
  
*select* \*  
*from* contract  
*where not* start\_date = '2022-01-01';  
  
*select* \*  
*from* contract  
*where not*(start\_date = '2022-01-01' *and* (salary >= 4000 *or* salary < 3000));  
  
*select* id, daily\_rent\_price / area *as* cost\_per\_meter  
*from* retail\_outlet  
*where* daily\_rent\_price / area > 3.5;  
  
*select* id, amount\_of\_payment, *month  
from* monthly\_payments  
*where month in* ('January', 'February', 'March', 'April');  
  
*select* \*  
*from* contract  
*where* start\_date *between DATE*('2022-02-01') *and DATE*('2023-01-02');  
  
*select* \*  
*from* employee  
*where* employee.last\_name *like* '%e';  
  
*select* \*  
*from* retail\_outlet  
*where* daily\_rent\_price *is null*;

*use* db\_labs;  
  
*select name*, employees\_amount - (*select* min(employees\_amount)  
 *from* shopping\_center) *as* diff\_from\_min  
*from* shopping\_center;  
  
*select* empl\_contr.first\_name, empl\_contr.last\_name, empl\_contr.salary - (*select* avg(salary)  
 *from* contract) *as* salary\_deviation  
*from* (*select* employee.first\_name, employee.last\_name, contract.salary  
 *from* employee, contract  
 *where* employee.id = contract.employee\_id) *as* empl\_contr;  
  
*select* id, *name*, address  
*from* shopping\_center  
*where exists*(  
 *select* shopping\_center\_id  
 *from* retail\_outlet  
 *where* shopping\_center.id = retail\_outlet.shopping\_center\_id  
 );  
  
*select* id, *name*, address  
*from* shopping\_center  
*where* id *in* (*select* shopping\_center\_id *from* retail\_outlet);  
  
*select* start\_date, end\_date, daily\_rent\_price  
*from* treaty, retail\_outlet  
*limit* 31;  
  
*select* first\_name, last\_name, payment\_card\_number, *month*, amount\_of\_payment  
*from client*, monthly\_payments  
*where client*.id = monthly\_payments.client\_id  
*order by* first\_name;  
  
*select* first\_name, last\_name, payment\_card\_number, *month*, amount\_of\_payment  
*from client*, monthly\_payments  
*where client*.id = monthly\_payments.client\_id *and* amount\_of\_payment < 300;  
  
*select* first\_name, last\_name, salary, position  
*from* employee  
*join* contract *on* employee.id = contract.employee\_id;  
  
*select* first\_name, last\_name, retail\_outlet\_id, sc.name, sc.address  
*from client  
left join* treaty *on client*.id = client\_id  
*left join* retail\_outlet *on* treaty.retail\_outlet\_id = retail\_outlet.id  
*left join* shopping\_center sc *on* retail\_outlet.shopping\_center\_id = sc.id;  
  
*select* first\_name, last\_name, retail\_outlet\_id, sc.name, sc.address  
*from client  
right join* treaty *on client*.id = client\_id  
*right join* retail\_outlet *on* treaty.retail\_outlet\_id = retail\_outlet.id  
*right join* shopping\_center sc *on* retail\_outlet.shopping\_center\_id = sc.id;  
  
*select* first\_name, last\_name, address  
*from client  
union select* first\_name, last\_name, address  
*from* employee;

**Висновок:**

На даній лабораторній роботі я вивчив та використав на практиці команди для вибірки даних з таблиці та команди для створення запитів з використанням підзапитів та з’єднань таблиці.