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 19

Виконав студент ІП-13 Нещерет Віталій Олександрович

(шифр, прізвище, ім'я, по батькові)

Перевірив Марченко Олена Іванівна

(прізвище, ім'я, по батькові)

Київ 2022

**Лабораторна робота №2**

**Варіант 19**

**Тема:** Побудова простих запитів

**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**

**Створені запити:**

SELECT \*  
FROM `комерційний банк`  
WHERE `банк id` = 2;  
  
SELECT \*  
FROM `реквізити`  
WHERE `сума` > 50000;  
  
SELECT \*  
FROM `реквізити`  
WHERE `сума` > 50000 AND `дата` > '2022-06-06';  
  
SELECT \*  
FROM `реквізити`  
WHERE `сума` > 50000 OR `дата` > '2022-06-06';  
  
SELECT \*  
FROM `реквізити`  
WHERE NOT `сума` > 50000;  
  
SELECT \*  
FROM `реквізити`  
WHERE NOT `сума` > 50000 AND `дата` > '2022-06-06';  
  
SELECT `кошторис id`, `ліміт витрат`, `витрачена сума`  
FROM `кошторис`  
WHERE `витрачена сума` / 2 <= 1000000;  
  
SELECT `кошторис id`, `ліміт витрат`, `витрачена сума`,  
 `ліміт витрат` - `витрачена сума` AS 'залишок'  
FROM `кошторис`;  
  
SELECT \*  
FROM `бюджетна установа`  
WHERE `бюджетна установа id` IN (2, 4, 8, 9);  
  
SELECT \*  
FROM `бюджетна установа`  
WHERE `бюджетна установа id` BETWEEN 2 AND 8;  
  
SELECT \*  
FROM `казначейство`  
WHERE `територіальна належність` LIKE '%na%';  
  
SELECT \*  
FROM `комерційний банк`  
WHERE `єдрпоу` IS NOT NULL;  
  
-- частина 2 --  
  
SELECT `територіальна належність`, `б у`.адреса, `б у`.назва  
FROM `казначейство` `к`  
JOIN `бюджетна установа` `б у` on `к`.`казначейство id` = `б у`.`казначейство id`;  
  
SELECT kk.`кошторис id`, kk.призначення, kk.рік  
FROM (SELECT `кекв`.`призначення`, кекв.`кошторис id`, `кошторис`.`рік`  
 FROM `кошторис`, `кекв`  
 WHERE кекв.`кошторис id` = кошторис.`кошторис id`) AS kk  
ORDER BY kk.`кошторис id`;  
  
SELECT \*  
FROM `комерційний банк`  
WHERE *EXISTS*(  
 SELECT \*  
 FROM `казначейство-банк`  
 WHERE `банк id`=`комерційний банк`.`банк id` AND `казначейство id` = 2  
 );  
  
SELECT \*  
FROM `кошторис`  
WHERE `кошторис id` IN (  
 SELECT `кошторис id`  
 FROM `затверджений кошторис`  
 );  
  
SELECT \*  
FROM казначейство  
CROSS JOIN `бюджетна установа`;  
  
SELECT pd.`доручення id`, pd.`затверджений казначейством`, r.дата, r.сума, r.номер, r.отримувач, r.підстава  
FROM `платіжне доручення` pd  
JOIN `реквізити` r on r.`реквізити id` = pd.`реквізити id`;  
  
SELECT pd.`доручення id`, pd.`затверджений казначейством`, r.дата, r.сума, r.номер, r.отримувач, r.підстава  
FROM `платіжне доручення` pd  
JOIN `реквізити` r on r.`реквізити id` = pd.`реквізити id` AND r.сума > (SELECT *AVG*(сума) FROM реквізити);  
  
SELECT б.`банк id`, `назва` AS 'назва банку', `єдрпоу`, `територіальна належність`  
FROM `комерційний банк` б  
INNER JOIN `казначейство-банк` `к-б` on б.`банк id` = `к-б`.`банк id`  
INNER JOIN казначейство к on `к-б`.`казначейство id` = к.`казначейство id`;  
  
SELECT б.`банк id`, `назва` AS 'назва банку', `єдрпоу`, `територіальна належність`  
FROM `комерційний банк` б  
LEFT OUTER JOIN `казначейство-банк` `к-б` on б.`банк id` = `к-б`.`банк id`  
LEFT OUTER JOIN казначейство к on `к-б`.`казначейство id` = к.`казначейство id`;  
  
SELECT б.`банк id`, `назва` AS 'назва банку', `єдрпоу`, `територіальна належність`  
FROM `комерційний банк` б  
RIGHT OUTER JOIN `казначейство-банк` `к-б` on б.`банк id` = `к-б`.`банк id`  
RIGHT OUTER JOIN казначейство к on `к-б`.`казначейство id` = к.`казначейство id`;  
  
SELECT `банк id` AS 'id закладу', `назва`  
FROM `комерційний банк`  
UNION (  
 SELECT `бюджетна установа id`, `назва`  
 FROM `бюджетна установа`  
);

**Висновок:**

На даній лабораторній роботі я використав команди для створення запитів в базах даних для отримання вибірки необхідних даних. Вивчив можливі способи об’єднання таблиць та дій над даними.