

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»**

**Інститут комп'ютерних наук та інформаційних технологій**

**Кафедра систем штучного інтелекту**



**Звіт до лабораторної роботи №5**

з дисципліни

“Організація Баз Даних та знань”

**Виконала:**

ст. гр. КН-210  
Заремба Вікторія

**Викладач:**

Мельникова Н.І.

## Лабораторна робота №5

з курсу “ОБДЗ”

на тему:

### **“Виконання теоретико-множинних операцій реляційної алгебри засобами SQL”**

**Мета роботи:** Розробити SQL запити для виконання операцій реляційної алгебри: об'єднання, перетину, різниці, декартового добутку

#### **Короткі теоретичні відомості.**

Реляційна алгебра – це множина операцій, що виконуються над відношеннями і мають за мету утворення нових відношень або їх станів. Реляційна алгебра визначає операції, які однаково чиним реалізуються в усіх базах даних реляційного типу, незалежно від їх змісту і технологій, за допомогою яких вони реалізовані. Тобто реляційна алгебра представляє собою процедурну мову обробки реляційних таблиць.

Реляційна алгебра складається з таких операцій: об'єднання, перетин, різниця, декартовий добуток, проекція, селекція, натуральне з'єднання, умовне з'єднання, а також операції включення/вилучення кортежу з відношень, включення/вилучення атрибуту з відношення, зміни параметрів атрибуту.

Перші чотири операції взяті з математичної теорії множин і практично співпадають з операціями над множинами. Це зручно, оскільки реляційні таблиці є множинами, і цілком природно застосовувати до них операції над множинами.

Об'єднанням двох відношень  $R$  та  $S$  з відповідними множинами атрибутів  $(A_1, A_2, \dots, A_n)$  називається відношення  $T$ , що має ту саму множину атрибутів  $(A_1, A_2, \dots, A_n)$ , а його інформаційне наповнення утворюється кортежами першого та другого відношень за вилученням повторень:

$$R \cup S = T(A_1, A_2, \dots, A_n) = \{r\} \cup \{s\}$$

Об'єднання дозволяє нам комбінувати дані з двох таблиць з однаковими множинами атрибутів. Однакові множини атрибутів потрібні для того, щоб результатом виконання операції об'єднання була реляційна таблиця.

Перетином двох відношень  $R$  та  $S$  з відповідними множинами атрибутів  $(A_1, A_2, \dots, A_n)$  називається відношення  $T$ , що має ту саму множину атрибутів  $(A_1, A_2, \dots, A_n)$ , а його інформаційне наповнення утворюється кортежами, які є спільними для цих двох відношень:

$$R \cap S = T(A_1, A_2, \dots, A_n) = \{r\} \cap \{s\}$$

Операція перетину дозволяє нам ідентифікувати рядки, спільні для двох таблиць.

Різницею двох відношень  $R$  та  $S$  з відповідними множинами атрибутів  $(A_1, A_2, \dots, A_n)$  називається відношення  $T$ , що має ту саму множину атрибутів  $(A_1, A_2, \dots, A_n)$ , а його інформаційне наповнення утворюється кортежами першого відношення за вилученням кортежів, які є спільними з другим відношенням:

$$R \setminus S = T(A_1, A_2, \dots, A_n) = \{r\} \setminus \{s\}$$

Операція різниці дозволяє ідентифікувати ті рядки, які є в одній таблиці, але відсутні в іншій.

Декартовим добутком двох відношень  $R$  та  $S$  з відповідними множинами атрибутів  $(A_1, A_2, \dots, A_n)$  та  $(B_1, B_2, \dots, B_m)$  називається нове відношення  $T$ , множина атрибутів якого складається з об'єднання множин атрибутів двох відношень, а кожен кортеж інформаційного наповнення утворюється шляхом конкатенації (сполучення) кожного кортежу першого відношення з кожним кортежем другого відношення.

Для реалізації теоретико-множинних операцій на мові SQL використовують директиву **SELECT**, спрощений опис якої наведено далі, а також функції роботи з множинами значень **IN()**, **NOT IN()**.

## SELECT

[ALL | DISTINCT | DISTINCTROW ]

*елемент\_вибірки* [, *елемент\_вибірки*]

[FROM *перелік\_таблиць*]

[WHERE *умова\_відбору*]

*елемент\_вибірки*

Вираз, або назва поля, значення якого потрібно вибрати. Символ «\*» позначає всі поля.

*перелік\_таблиць*

Назва таблиці, з якої здійснюється вибір значень.

*умова\_відбору*

Вказує умови відбору потрібних записів.

DISTINCT | DISTINCTROW

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

Для того, щоб виконати операцію об'єднання таблиць, потрібно за допомогою команди

**UNION** об'єднати результати вибору рядків з двох, або більше, таблиць. Наведемо синтаксис команди.

SELECT ...

**UNION** [ALL | DISTINCT] SELECT ...

[UNION [ALL | DISTINCT] SELECT ...]

## Хід роботи

Виконаємо операції об'єднання, перетину, різниці і декартового добутку. Для цього потрібно створити 2 таблиці з однаковими множинами атрибутів. В нашій базі даних є таблиця user. На її основі створимо таблиці User1 s User2 (так, щоб вони перетинались)

```
CREATE TABLE pm_system.user1
AS SELECT id_user, surname, name FROM user
WHERE id_user<8;

CREATE TABLE pm_system.user2
AS SELECT id_user, surname, name FROM user
WHERE id_user>5;
```

	id_user	surname	name
1	1	Zaremba	Viktor
2	2	Vyshnevskaya	Anastasia
3	3	Boyechko	Denys
4	5	Kondratyuk	Bogdana
5	6	Rogynski	Roman
6	7	Osovskaya	Sophia

Таблиця User1

	id_user	surname	name
1	6	Rogynski	Roman
2	7	Osovskaya	Sophia
3	8	Mačukiv	Nastya
4	9	Korin	Olya
5	10	Ostashevskaya	Nadiya
6	11	Drahovik	Yarina
7	12	Zubik	Vika

Таблиця User2

### 1. Виконуємо запит операції об'єднання user1 I user2.

```
SELECT * FROM user1
UNION SELECT * FROM user2;
```

Результат виконання операції

	id_user	surname	name
1	1	Zaremba	Viktor
2	2	Vyshnevskaya	Anastasia
3	3	Boyechko	Denys
4	5	Kondratyuk	Bogdana
5	6	Rogynski	Roman
6	7	Osovskaya	Sophia
7	8	Mačukiv	Nastya
8	9	Korin	Olya
9	10	Ostashevskaya	Nadiya
10	11	Drahovik	Yarina
11	12	Zubik	Vika

### 2. Запит на виконання перетину.

```
SELECT * FROM user1
WHERE id_user IN (SELECT id_user from user2);
```

Результат виконання операції

	id_user	surname	name
1	6	Rogynski	Roman
2	7	Osovskaya	Sophia

### 3. Запит на виконання різниці user1 I user2.

```
SELECT * FROM user1
WHERE id_user NOT IN (SELECT id_user from user2);
```

Результат виконання операції:

	id_user	surname	name
1	1	Zaremba	Viktoria
2	2	Vyshnevskaya	Anastasia
3	3	Boyechko	Denys
4	5	Kondratyuk	Bogdana

### 4. Запит на виконання декартового добутку двох таблиць.

```
SELECT * FROM user1, user2;
```

Результат виконання операції:

	user1.id_user	user1.surname	user1.name	user2.id_user	user2.surname	user2.name
1	1	Zaremba	Viktoria	6	Rogynski	Roman
2	2	Vyshnevskaya	Anastasia	6	Rogynski	Roman
3	3	Boyechko	Denys	6	Rogynski	Roman
4	5	Kondratyuk	Bogdana	6	Rogynski	Roman
5	6	Rogynski	Roman	6	Rogynski	Roman
6	7	Osovskaya	Sophia	6	Rogynski	Roman
7	1	Zaremba	Viktoria	7	Osovskaya	Sophia
8	2	Vyshnevskaya	Anastasia	7	Osovskaya	Sophia
9	3	Boyechko	Denys	7	Osovskaya	Sophia
10	5	Kondratyuk	Bogdana	7	Osovskaya	Sophia
11	6	Rogynski	Roman	7	Osovskaya	Sophia
12	7	Osovskaya	Sophia	7	Osovskaya	Sophia
13	1	Zaremba	Viktoria	8	Maцkiv	Nastya
14	2	Vyshnevskaya	Anastasia	8	Maцkiv	Nastya
15	3	Boyechko	Denys	8	Maцkiv	Nastya
16	5	Kondratyuk	Bogdana	8	Maцkiv	Nastya
17	6	Rogynski	Roman	8	Maцkiv	Nastya
18	7	Osovskaya	Sophia	8	Maцkiv	Nastya
19	1	Zaremba	Viktoria	9	Korin	Olya
20	2	Vyshnevskaya	Anastasia	9	Korin	Olya
21	3	Boyechko	Denys	9	Korin	Olya
22	5	Kondratyuk	Bogdana	9	Korin	Olya
23	6	Rogynski	Roman	9	Korin	Olya
24	7	Osovskaya	Sophia	9	Korin	Olya
25	1	Zaremba	Viktoria	10	Ostashevskaya	Nadiya
26	2	Vyshnevskaya	Anastasia	10	Ostashevskaya	Nadiya
27	3	Boyechko	Denys	10	Ostashevskaya	Nadiya
28	5	Kondratyuk	Bogdana	10	Ostashevskaya	Nadiya
29	6	Rogynski	Roman	10	Ostashevskaya	Nadiya

**Висновок:** на цій лабораторній роботі було розглянуто операції реляційної алгебри та їх реалізація на мові SQL. Здійснено об'єднання, перетин, різницю та декартовий добуток двох таблиць.