МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

КАФЕДРА СИСТЕМ ШТУЧНОГО ІНТЕЛЕКТУ



Звіт до лабораторної роботи №5

з дисципліни:

“ОБДЗ”

на тему:

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

**Підготувала:**

студентка групи КН-209

Дипко Олександра

**Викладач:**

Мельникова Н.І.

Львів 2020

**Мета роботи:**

виконати теоретико-множинних операцій реляційної алгебри засобами SQL.

**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 однаковим чином 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 ,..., *An* )

називається відношення *T* , що має ту саму множину атрибутів ( *A*1 , *A*2 ,..., *An* ) , а

його інформаційне наповнення утворюється кортежами першого та другого відношень за вилученням повторень:

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 ,..., *An* )

називається відношення *T* , що має ту саму множину атрибутів ( *A*1 , *A*2 ,..., *An* ) , а

його інформаційне наповнення утворюється кортежами, які є спільними для цих двох відношень:

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

*Різницею* двох відношень *R* та *S* з відповідними множинами атрибутів

( *A*1 , *A*2 ,..., *An* )

називається відношення *T* , що має ту саму множину атрибутів

( *A*1 , *A*2 ,..., *An* ) , а його

інформаційне наповнення утворюється кортежами першого відношення за вилученням кортежів, які є спільними з другим відношенням:

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 ,..., *An* ) та (*B*1 , *B*2 ,..., *Bm* )

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**

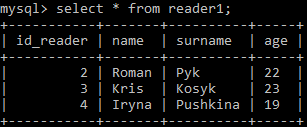
Перед виконанням завдання, потрібно сформувати дві таблиці з однаковими множинами атрибутів. Візьмемо за основу таблицю користувачів **reader** і виконаємо вибір двох множин записів, які перетинаються. Результат збережемо в таблицях **reader1** і **reader2**.

1. Виконати запит на виконання об’єднання **reader1** i **reader2**.
2. Виконати запит на виконання перетину **reader1** i **reader2**.
3. Запит на виконання різниці **reader1** i **reader2**.
4. Запит на виконання декартового добутку двох таблиць.

**CREATE TABLE** reader1

**AS SELECT** id\_reader, name, surname, age **FROM** reader

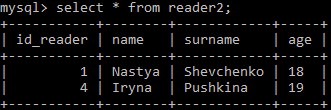
**WHERE** age > "18";



**CREATE TABLE** reader2

**AS SELECT** id\_reader, name , age **FROM** library.reader

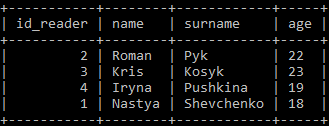
**WHERE** age < "22";



1. Виконати запит на виконання об’єднання **reader1** i **reader2.**

**SELECT** \* FROM reader1

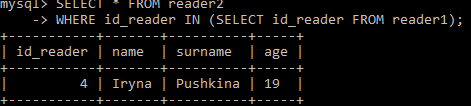
**UNION SELECT** \* FROM reader2



1. Виконати запит на виконання перетину **reader1** i **reader2**.

**SELECT** \* FROM reader2

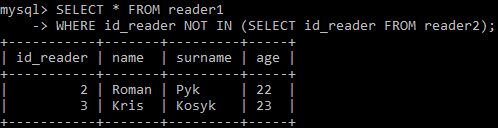
**WHERE** id\_reader **IN** (**SELECT** id\_reader **FROM** reader1);



1. Запит на виконання різниці **reader1** i **reader2**.

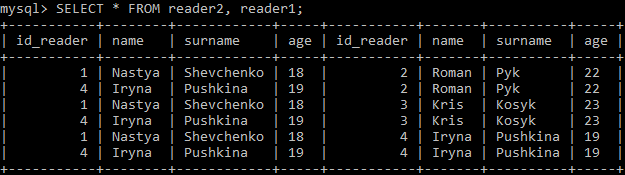
**SELECT** \* FROM reader1

**WHERE** id\_reader **NOT IN** (**SELECT** id\_reader **FROM** reader2);



1. Запит на виконання декартового добутку двох таблиць.

**SELECT** \* FROM reader2, reader1**;**



**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.