

Харківський національний університет імені В. Н. Каразіна  
Факультет комп'ютерних наук

ПРАКТИЧНА РОБОТА №6

на тему: «3. Магазин курток "Winter&Autumn"»

ВАРІАНТ 16

Виконала: студентка 2 курсу групи КС-22

Спеціальності 122 «Комп'ютерні науки»

Рижко Анна Дмитрівна

Прийняв: к.т.н., доцент каф. ММ та АД

Бережний Артем

Харків 2024

**Тема:** Реляційна алгебра.

**Мета роботи:** Набути досвіду створення запитів у реляційній моделі баз даних, ознайомитися з основними операціями реляційної алгебри та навчитися застосовувати їх для оптимізації запитів і роботи з даними.

### **Завдання №1.**

За своїм варіантом (додаток А) виконати завдання по реляційній алгебрі з побудовою відповідної структури запиту мовою реляційної алгебри.

Варіант 16

1. Отримати повну інформацію про постачальників
2. Отримати повну інформацію про всі проекти в Києві
3. Отримати всі поєднання "ім'я постачальника-статус"
4. Отримати всі кольори товарів
5. Отримати всі поставки, де кількість товару постачальника S2 більше 200
6. Отримати номери та імена постачальників з Києва
7. Отримати імена товарів синього кольору, які постачаються постачальником S3.
8. Отримати імена постачальників та їх статус, які беруть участь у проекті Офіси.
9. Отримати номери постачальників, які постачають або товари з Одеси або для проекту з Харкова.

#### **1) Отримати повну інформацію про постачальників**

Запит:

S

Пояснення:

Запит просто повертає всі дані з відношення "Постачальники" (S). Усі атрибути постачальника будуть включені в результат.

#### **2) Отримати повну інформацію про всі проекти в Києві**

Запит:

**J WHERE CITY='Київ'**

Пояснення:

Тут вибираються всі проекти (J), де значення атрибута CITY дорівнює 'Київ'. Це фільтрація проектів, які реалізуються в Києві.

### **3) Отримати всі поєднання "ім'я постачальника-статус"**

Запит:

**P [SHAME, STATUS]**

Пояснення:

Цей запит вибирає атрибути SNAME (ім'я постачальника) та STATUS (статус постачальника) з відношення "Постачальники" (S). Це дозволяє отримати всі можливі пари імен постачальників і їх статусів.

### **4) Отримати всі кольори товарів**

Запит:

**P [COLOR]**

Пояснення:

Цей запит вибирає всі значення атрибута COLOR з відношення "Товар" (P). Таким чином, ми отримуємо список всіх доступних кольорів товарів.

### **5) Отримати всі поставки, де кількість товару постачальника S2 більше 200**

Запит:

**(SPJ WHERE S#='S2' AND QUANTITY > 200)**

Пояснення:

Запит вибирає всі записи з відношення "Поставка" (SPJ), де номер постачальника дорівнює 'S2' і кількість товару перевищує 200 одиниць. Це дозволяє отримати всі поставки, що відповідають заданим умовам.

### **6) Отримати номери та імена постачальників з Києва**

Запит:

**(S WHERE CITY='Київ') [S#, SNAME]**

Або докладніше:

Фільтруємо постачальників (S), вибираючи лише тих, хто знаходиться в Києві:

**T1 := S WHERE CITY='Київ';**

Вибираємо атрибути S# (номер постачальника) та SNAME (ім'я постачальника) з отриманої таблиці:

**T2 := T1 [S#, SNAME];**

Пояснення:

Спочатку вибираємо з відношення "Постачальники" (S) тільки тих постачальників, які знаходяться в Києві (місто = 'Київ'). Потім проекція дає тільки номери постачальників (S#) та їх імена (SNAME).

**7) Отримати імена товарів синього кольору, які постачаються постачальником S3.**

Запит:

**((P WHERE COLOR='Синій') JOIN SPJ) WHERE S#='S3') [PNAME]**

Або докладніше:

Вибираємо всі товари, що мають колір "Синій":

**T1 := P WHERE COLOR='Синій';**

З'єднуємо отриману таблицю з таблицею "Поставка" (SPJ) за номером товару (P#):

**T2 := T1 JOIN SPJ;**

Фільтруємо записи, де номер постачальника дорівнює 'S3':

**T3 := T2 WHERE S#='S3';**

Вибираємо лише імена товарів (PNAME) з отриманої таблиці:

**T4 := T3 [PNAME];**

Пояснення:

Спочатку вибираємо з відношення "Товар" (P) ті товари, які мають колір 'Синій'. Потім виконуємо з'єднання з "Поставкою" (SPJ) за номером товару (P#). Після цього фільтруємо записи по постачальнику

з номером 'S3' (S#='S3'). Результатом буде проекція по іменах товарів (PNAME).

**8) Отримати імена постачальників та їх статус, які беруть участь у проекті Офіси.**

Запит:

**((SPJ JOIN J WHERE JNAME='Офіси') JOIN S) [SNAME, STATUS]**

Або докладніше:

Спочатку фільтруємо проекти, вибираючи проект з назвою "Офіси":

**T1 := J WHERE JNAME='Офіси';**

З'єднуємо таблицю "Поставка" (SPJ) з проектами (J) за номером проекту (J#):

**T2 := SPJ JOIN T1;**

З'єднуємо отриману таблицю з постачальниками (S) за номером постачальника (S#):

**T3 := T2 JOIN S;**

Вибираємо імена постачальників (SNAME) та їх статус (STATUS) з таблиці:

**T4 := T3 [SNAME, STATUS];**

Пояснення:

Спочатку з'єднуємо "Поставки" (SPJ) з проектами (J) за номером проекту (J#), фільтруючи проекти, де ім'я проекту дорівнює "Офіси" (JNAME='Офіси'). Потім здійснюємо з'єднання з постачальниками (S) за номером постачальника (S#). Результатом є імена постачальників (SNAME) і їх статус (STATUS), які беруть участь у проекті "Офіси".

**9) Отримати номери постачальників, які постачають або товари з Одеси або для проекту з Харкова.**

Запит:

**(SPJ JOIN (S WHERE CITY='Одеса')) [S#] UNION (SPJ JOIN (J WHERE CITY='Харків')) [S#]**

Або докладніше:

Спочатку вибираємо постачальників, що знаходяться в Одесі:

**T1 := S WHERE CITY='Одеса';**

З'єднуємо таблицю "Поставка" (SPJ) з таблицею постачальників (S) за номером постачальника (S#):

**T2 := SPJ JOIN T1;**

Проекцією вибираємо лише номери постачальників (S#):

**T3 := T2 [S#];**

Тепер для другої частини запиту:

Спочатку вибираємо проекти, що знаходяться в Харкові:

**T4 := J WHERE CITY='Харків';**

З'єднуємо таблицю "Поставка" (SPJ) з таблицею проектів (J) за номером проекту (J#):

**T5 := SPJ JOIN T4;**

Проекцією вибираємо лише номери постачальників (S#):

**T6 := T5 [S#];**

Об'єднуємо результати обох частин через оператор UNION, щоб отримати унікальні номери постачальників:

**T7 := T3 UNION T6;**

Пояснення:

- Першу частину запиту виконуємо, вибираючи постачальників з Одеси, потім з'єднуємо їх з "Поставками" (SPJ) за номером постачальника (S#). Проекція дає тільки номери постачальників (S#).
- Другу частину запиту виконуємо, вибираючи проекти, що знаходяться в Харкові (CITY='Харків'), потім з'єднуємо їх з "Поставками" (SPJ). Знову проекція дає номери постачальників (S#).
- Об'єднуємо результати обох частин через оператор UNION, що дає унікальні номери постачальників, які відповідають хоча б одній з умов.

## Завдання №2.

Для одного з запитів Вашого варіанту (основного, по якому були виконані практичні роботи 3, 4, 5) виконати побудову запиту до предметної області мовою реляційної алгебри.



Рисунок 1 – Результуюча діаграма магазину курток "Winter&Autumn"

### Запит:

Отримати імена всіх клієнтів, які зробили замовлення на чорні куртки на суму не менше 10000 грн. у період з 5 листопада по 5 грудня.

### Виконаний запит:

Вибрати товари, які є "куртками" чорного кольору з таблиці Товар:

**T1 := Товар WHERE Назва='куртка' AND Колір='чорний'**

Знайти всі строки замовлень, які містять ці товари, використовуючи Код Товару:

**T2 := Строка\_замовлення JOIN T1 ON  
Строка\_замовлення.Код\_Товару = T1.Код\_Товару**

Знайти замовлення, які відповідають строкам замовлення, використовуючи Код Замовлення:

**T3 := Замовлення JOIN T2 ON Замовлення.Код\_Замовлення =  
T2.Код\_Замовлення**

Обмежити замовлення до тих, що мають загальну вартість  $\geq 10,000$  грн і датуються між 5 листопада та 5 грудня:

**T4 := T3 WHERE Загальна\_вартість  $\geq 10000$  AND  
Дата\_замовлення  $\geq$  '2023-11-05' AND Дата\_замовлення  $\leq$  '2023-12-05'**

Отримати відповідних клієнтів, з'єднавши результати із таблицею Клієнт за Код Клієнта:

**T5 := Клієнт JOIN T4 ON Клієнт.Код\_Клієнта = T4.Код\_Клієнта**

Виконати проєкцію, щоб отримати лише ПІБ клієнтів:

**T6 := T5 [ПІБ\_Клієнта]**

Пояснення:

1. **(Товар WHERE Назва = 'куртка' AND Колір = 'чорний'):**  
Вибираємо чорні куртки з таблиці Товар.
2. **JOIN Строка\_замовлення ON Товар.Код\_Товару =  
Строка\_замовлення.Код\_Товару:**  
З'єднуємо чорні куртки зі строками замовлення (Строка\_замовлення) за Код Товару.
3. **JOIN Замовлення ON Строка\_замовлення.Код\_Замовлення =  
Замовлення.Код\_Замовлення:**  
З'єднуємо строки замовлення з таблицею Замовлення за Код Замовлення.
4. **WHERE Замовлення.Загальна\_вартість  $\geq 10000$ :**  
Вибираємо лише ті замовлення, де загальна вартість дорівнює або перевищує 10,000 грн.
5. **AND Замовлення.Дата\_замовлення  $\geq$  '2023-11-05' AND  
Замовлення.Дата\_замовлення  $\leq$  '2023-12-05':**  
Додатково фільтруємо замовлення, зроблені в період з 5 листопада по 5 грудня.
6. **JOIN Клієнт ON Замовлення.Код\_Клієнта = Клієнт.Код\_Клієнта:**  
З'єднуємо замовлення з таблицею Клієнт, щоб отримати дані про клієнтів.
7. **[ПІБ Клієнта]:**  
Виконуємо проєкцію, щоб залишити тільки імена клієнтів, які відповідають критеріям.



**Висновки:**

У ході виконання роботи було досягнуто основної мети: набуття практичних навичок створення запитів у реляційній моделі баз даних. Було успішно застосовано основні операції реляційної алгебри, такі як вибірка, проекція, з'єднання та фільтрація даних. Це дозволило побудувати ефективний запит для вирішення практичної задачі — отримання даних про клієнтів, які виконали замовлення за конкретними умовами.

**Посилання на GitHub:**

**<https://github.com/Ankkaa15/Pr-1>**