



ТЕМА 2. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ЗБЕРІГАННЯ ДАНИХ В ІАС

ЗАНЯТТЯ 7. ОСНОВИ ВИКОРИСТАННЯ ЗАСОБІВ ВИБОРУ ІНФОРМАЦІЇ З БД. ПОБУДОВА ЗАПИТІВ

НАВЧАЛЬНІ ПИТАННЯ:

1. Класифікація команд SQL та місце DML.
2. Структура та порядок виконання команди SELECT.
3. Фільтрація, сортування та агрегація даних для аналітичних звітів.
4. Об'єднання таблиць та оптимізація запитів.

МЕТА ЗАНЯТТЯ:

Опанувати синтаксис SQL для отримання оперативних даних з ІАС (Інформаційно-аналітичної системи) для підтримки прийняття управлінських рішень.

АКТУАЛЬНІСТЬ ТЕМИ: ДАНІ ЯК ЗБРОЯ

Хаос даних



SQL QUERY

🔍 Ручний перебір, затримка ⏱️ часу, неточність. 🕒 ?

Оперативна обізнаність

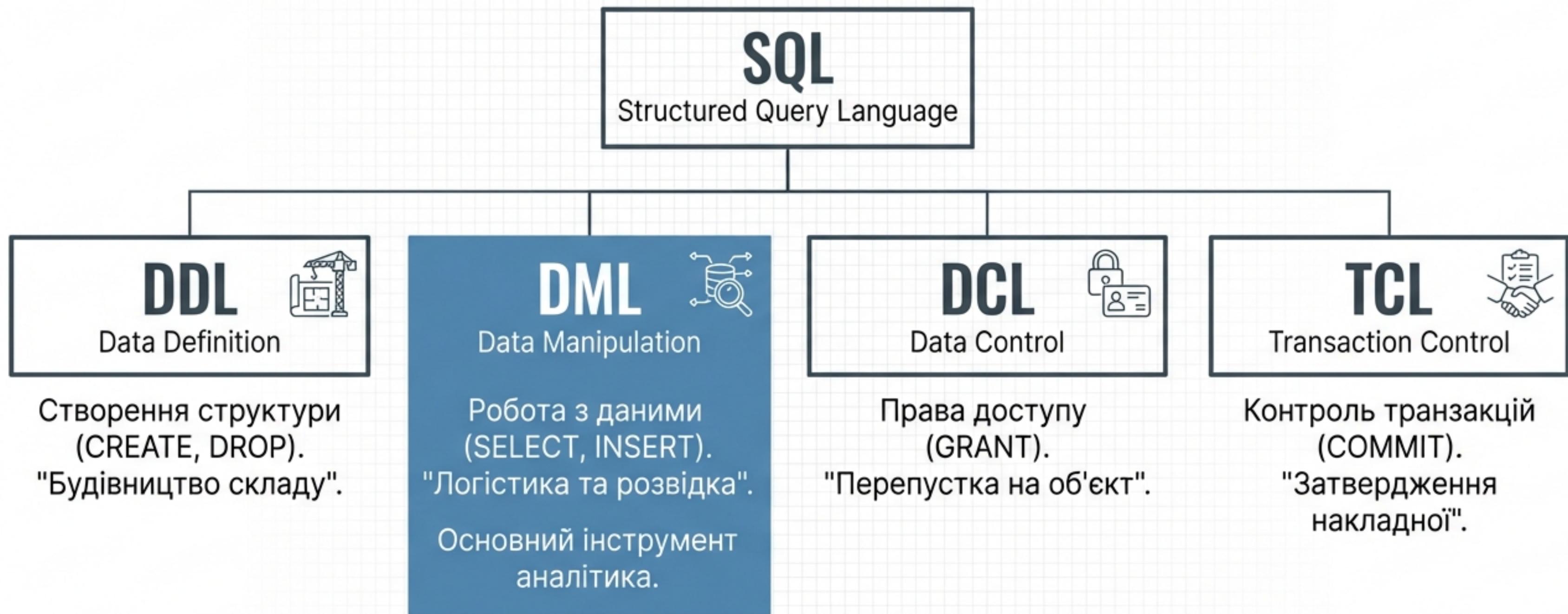
OBJECT_ID	LOCATION_LAT	LOCATION_LONG	STATUS	UNIT_TYPE	CRITICAL_VALUE
1	37.92040	-18.61536	ALERT	UNIT	8.05
2	37.93833	-18.61238	ALERT	UKIT	500
3	37.95217	-18.61236	ALERT	UNIT	500
4	37.72157	-18.63237	HIGH-PRIORITY	TARGET	8.180
5	37.25362	-18.61295	ALERT	UKIT	8.50
7	37.72050	-18.63925	ALERT	UNIT	1.890
8	37.95385	-18.61978	ALERT	UNIT	1600
10	37.92398	-18.61225	ALERT	UKIT RESOURCE	2.5
11	37.75294	-18.63745	ALERT	UNIT	160
12	37.73385	-18.63042	ALERT	UKIT	180
...

⚡️ 💾🎯 Миттєвий доступ, єдине джерело правди, точність.

КЛЮЧОВА ТЕЗА:

Якість управлінського рішення командира прямо залежить від швидкості та коректності вибірки даних з ІАС. **Помилка в запиті = помилка в плануванні.**

КЛАСИФІКАЦІЯ SQL: МІСЦЕ DML

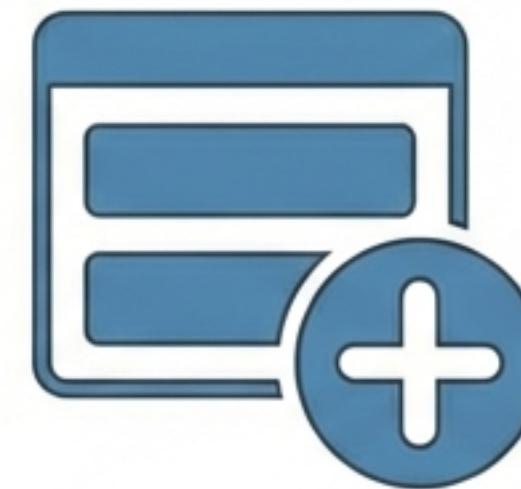


ОСНОВНІ КОМАНДИ DML



SELECT

Вибірка даних. Читання інформації без зміни вмісту БД. Основна команда розвідки.



INSERT

Вставка нових рядків. Введення нових бойових донесень або ресурсів.



UPDATE

Оновлення існуючих даних. Зміна статусу техніки або особового складу.



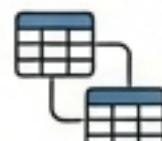
DELETE

Видалення записів. Знищення або архівація застарілих даних.

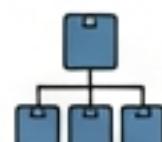
АНАТОМІЯ ТА ПОРЯДОК ВИКОНАННЯ SELECT

ПОРЯДОК НАПИСАННЯ

 **SELECT** (Які поля?)

 **FROM** (З якої таблиці?)

 **WHERE** (Який фільтр?)

 **GROUP BY** (Як групувати?)

 **HAVING** (Фільтр груп?)

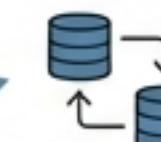
 **ORDER BY** (Як сортувати?)

 **LIMIT** (Скільки рядків?)

ЛОГІКА ВИКОНАННЯ СИСТЕМОЮ

 **FROM** (Джерело даних)

 **WHERE** (Відсів рядків)

 **GROUP BY** (Агрегація)

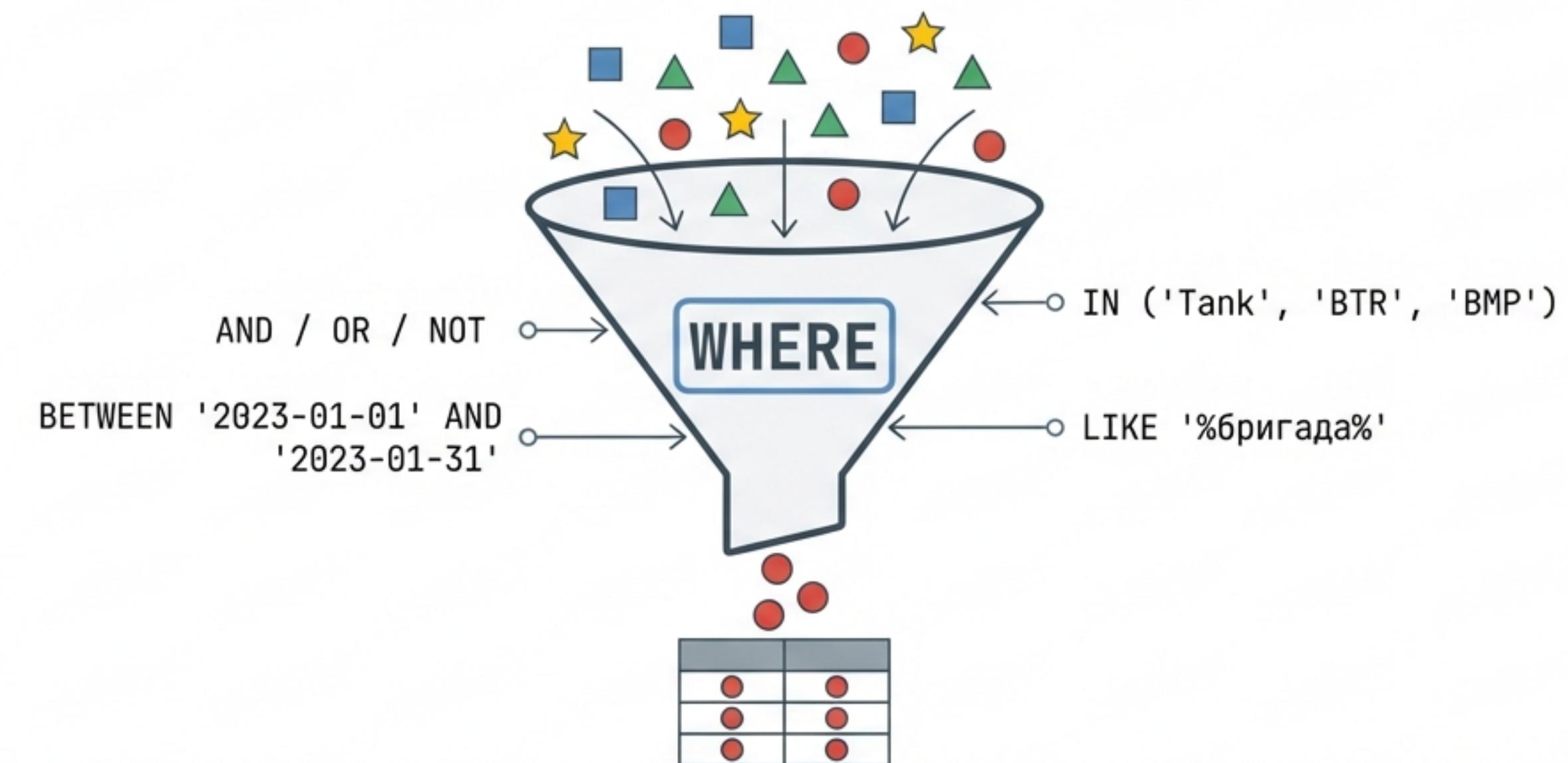
 **HAVING** (Відсів груп)

 **SELECT** (Формування проекції)

 **ORDER BY** (Сортування)

 **LIMIT** (Обрізка)

ФІЛЬТРАЦІЯ ДАНИХ: ОПЕРАТОР WHERE

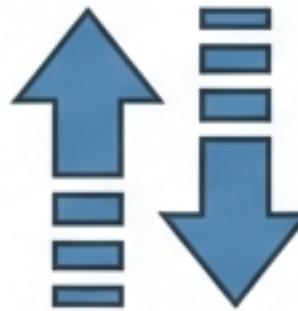


Приклад: Знайти активну техніку з малим запасом пального.

```
SELECT unit_name, fuel_level  
FROM vehicles  
WHERE status = 'active' AND fuel_level < 30;
```

СОРТУВАННЯ ТА ОБМЕЖЕННЯ (ORDER BY / LIMIT)

ORDER BY (Сортування)



- ASC (За зростанням): Від А до Я, 0-9 (за замовчуванням).
- DESC (За спаданням): Від Я до А, 9-0. Використовується для виведення пріоритетних даних.

LIMIT (Обмеження)



- Відсікання зайвих рядків для отримання ТОП-списків.

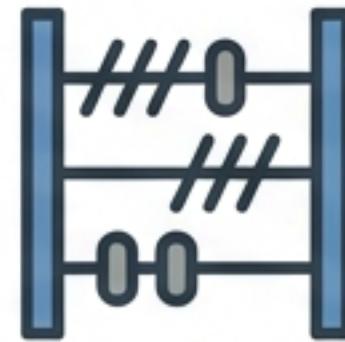
Результат запиту (ТОП-3 втрат):

Підрозділ	Втрати (од.)
1-й Батальон	12
3-й Батальон	8
2-й Батальон	5

← ORDER BY losses DESC LIMIT 3

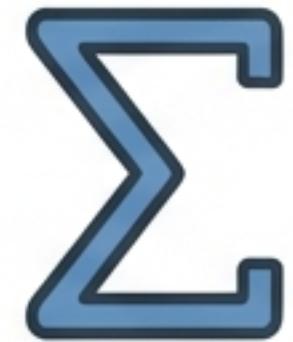
АГРЕГАТНІ ФУНКЦІЇ: СИНТЕЗ ДАНИХ

COUNT(*)



Кількість рядків.
(Особовий склад).

SUM()



Сума значень.
(Загальний обсяг БК).

AVG()



Середнє арифметичне.
(Середня укомплектованість).

MAX()



Максимум.
(Найбільша дальності).

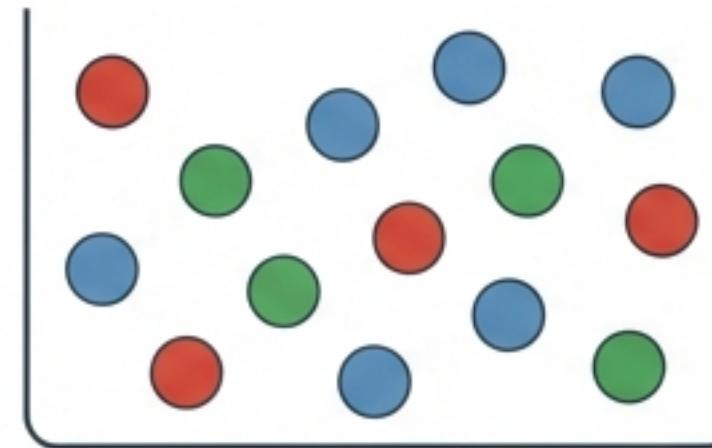
MIN()



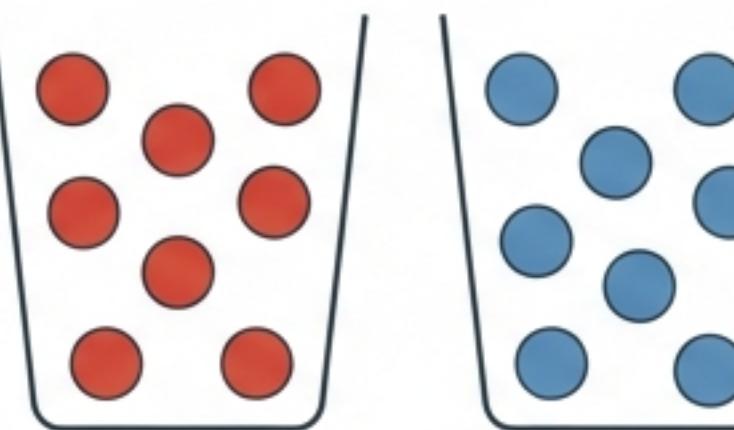
Мінімум.
(Критичний залишок).

ГРУПУВАННЯ: GROUP BY ТА HAVING

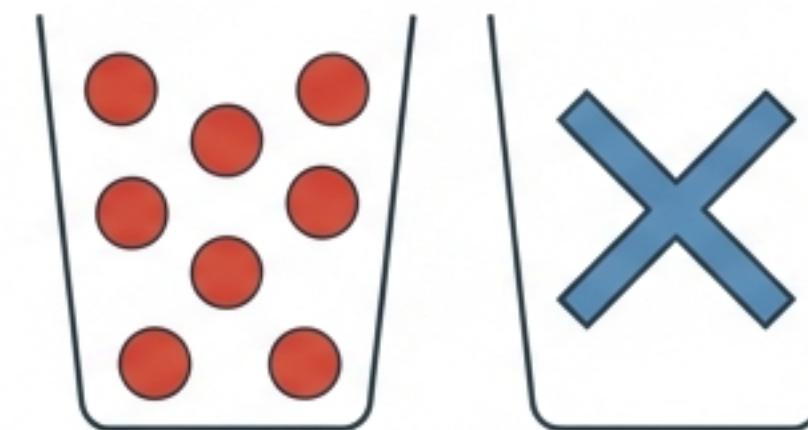
Raw Data



Grouped Data



Filtered Groups



`>_ WHERE color != 'Green'`
(Filter rows)

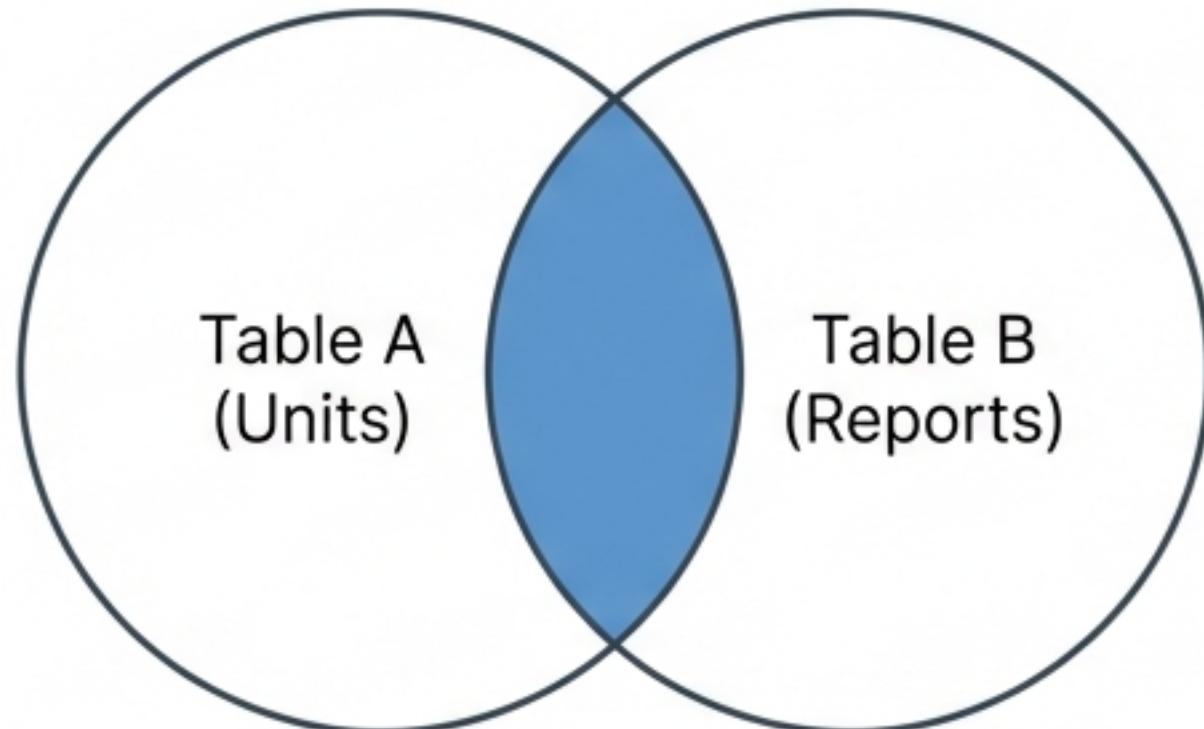
→ GROUP BY color
(Aggregate into buckets)

→ HAVING count > 5
(Filter entire groups)

ВАЖЛИВО: WHERE фільтрує окремі рядки **ПЕРЕД** групуванням.
HAVING фільтрує результати **ПІСЛЯ** агрегації.

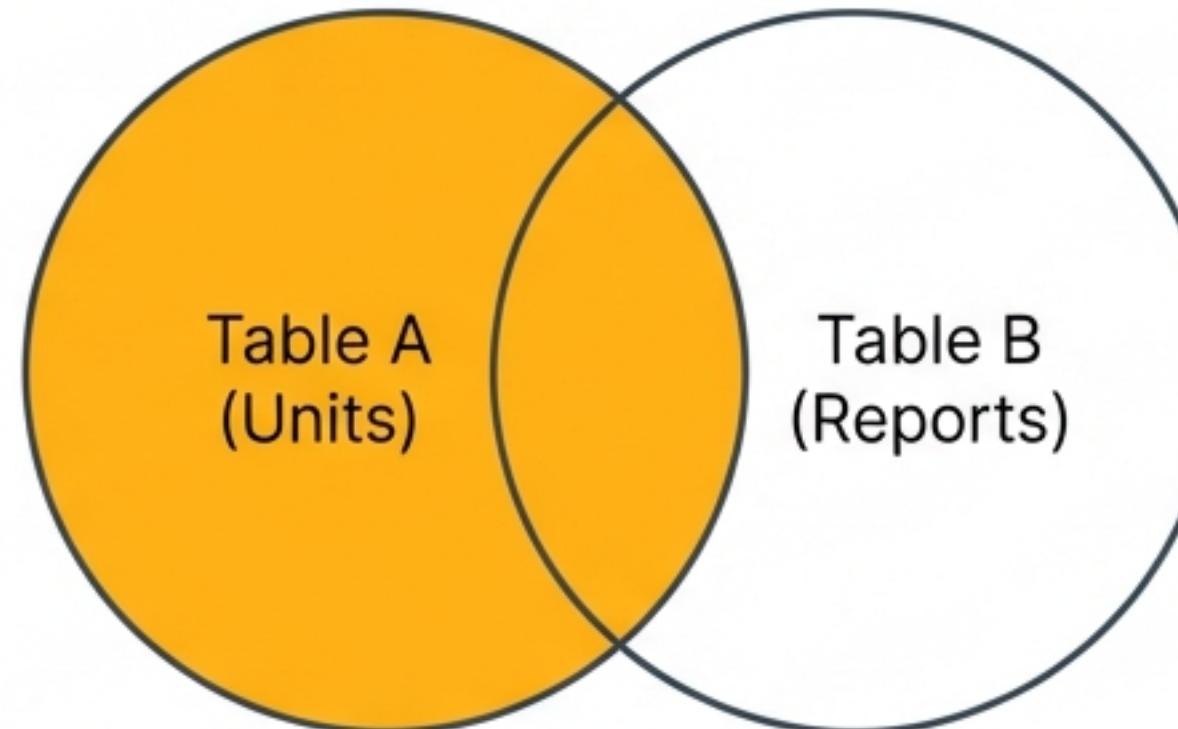
ОБ'ЄДНАННЯ ТАБЛИЦЬ (JOIN)

INNER JOIN



Перетин. Тільки підрозділи, які подали звіти.

LEFT JOIN



Лівостороннє. Всі підрозділи, навіть ті, що НЕ подали звіти (NULL).

Критично для виявлення 'мовчунів'.

```
>  
FROM units u  
LEFT JOIN reports r ON u.id = r.unit_id
```

АЛГОРИТМ ПОБУДОВИ ЗАПИТУ (SOP)



ТИПОВІ АНАЛІТИЧНІ ЗАДАЧІ В IAC



МІСІЯ (Mission)

Виявити підрозділи, що не надіслиали ранкове донесення.



SQL СТРАТЕГІЯ (SQL Strategy)



```
LEFT JOIN ...  
WHERE report.id IS NULL
```



```
GROUP BY sector ...  
ORDER BY attacks DESC LIMIT 3
```



```
WHERE date BETWEEN (current_date  
- 7) AND current_date
```



```
AVG(weight_tons)  
GROUP BY logistic_hub
```

Проаналізувати динаміку витрат пального за останній тиждень.

Розрахувати середнє навантаження на логістичний вузол.

ТИПОВІ ПОМИЛКИ (“FRIENDLY FIRE”)



Cartesian Product

Відсутність умови 'ON' у JOIN. Призводить до перемноження всіх рядків і зависання системи.



Logic Trap

Плутанина з дужками при використанні AND та OR.
 $(A \text{ AND } B) \text{ OR } C \neq A \text{ AND } (B \text{ OR } C)$.



Aggregate in WHERE

Спроба написати 'WHERE sum(x) > 10'. Для агрегатів існує тільки HAVING.



SELECT *

Запит всіх колонок у великих таблицях. Перевантаження каналу зв'язку.

ОПТИМІЗАЦІЯ ЗАПИТІВ



Економія полів

Обирати тільки конкретні колонки (`SELECT name, id`), а не все підряд.



Використання індексів

Фільтрувати (`WHERE`) по індексованих полях (ID, дати, коди) для уникнення повного сканування.



EXPLAIN

Команда розвідки запиту. Дозволяє побачити план виконання та вартість операції до її запуску.



Мінімізація підзапитів

Використовувати `JOIN` замість складних вкладених конструкцій де це можливо.



ПІДСУМКИ ТА ВИСНОВКИ



SQL — це мова оперативного управління. Це інструмент ситуаційної обізнаності, а не просто програмування.

Точність запиту = Точність рішення.
Помилка у вибірці може спотворити оцінку обстановки.

Швидкість та Оптимізація. В умовах слабких каналів зв'язку ефективний код є критичним.

НАСТУПНЕ ЗАНЯТТЯ: Практичне відпрацювання нормативів з написання пошукових запитів.