Міністерство освіти і науки України НТУУ «Київський політехнічний інститут» Фізико-технічний інститут

Системи та засоби інтерактивної аналітики

Лабораторна робота №4

Advanced SQL

Варіант №6

Виконав: Студент 4 курсу ФТІ групи ФБ-01

Сахній Назар

Перевірив:

Лабораторна робота №4 Advanced SQL

Тема: Запити розширеного SQL (advanced SQL).

Завдання: Навчитися створювати розширені SQL запити.

№ варіанта	Завдання
6	Використати існуючу БД (лаб. роб. 2,3) Запити: 1. Вивести повні адреси ЈЅ скриптів (розширення јѕ) і їх розмір, які запросили з ком' пютера з ІР адресою 83.227.29.211, для яких запит завершився вдало. 2. Придумайте будь який запит з використанням UNION. 3. Знайдіть розмір всього скачаного комп'ютером з ІР адресою 83.227.29.211

Робоче завдання

1. До БД створеної у попередніх роботах імпортувати таблицю з даними.



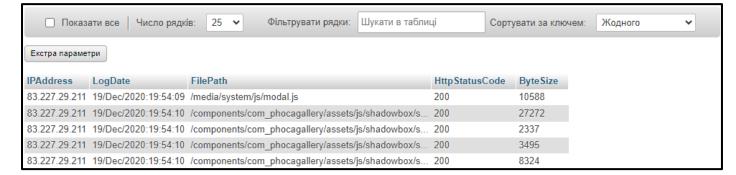


2. Написання відповідних SQL-запитів, які необхідно для виконання завдання.

2.1. Вивести повні адреси JS скриптів (розширення **js**) і їх розмір, які запросили з комп'ютера з IP адресою **83.227.29.211**, для яких запит **завершився вдало**:

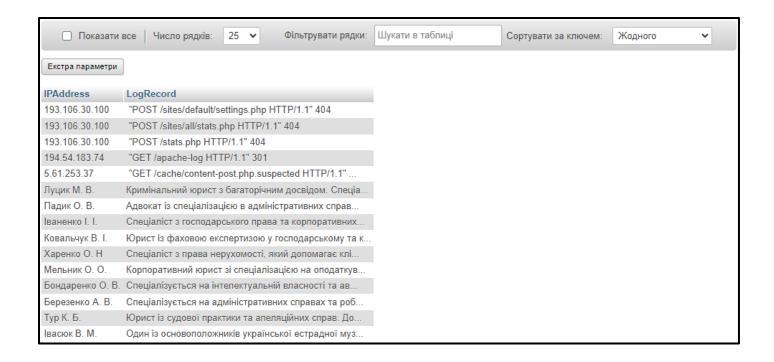
```
Bukohatu SQL sanut(u) y 6asi даних law_firm_pro.tblaccesslog: (a)

1 SELECT
2 SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' ', 1), ' ', -1) AS IPAddress,
3 SUBSTRING_INDEX(SUBSTRING_INDEX(Line, '[', -1), ' ', 1) AS LogDate,
4 SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' ', 7), ' ', -1) AS FilePath,
5 SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' ', 9), ' ', -1) AS HttpStatusCode,
6 SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' ', 10), ' ', -1) AS ByteSize
7 FROM
8 tblaccesslog
9 WHERE
10 Line LIKE ('83.227.29.211%.js%')
AND SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' ', 9), ' ', -1) BETWEEN '200' AND '206'
```



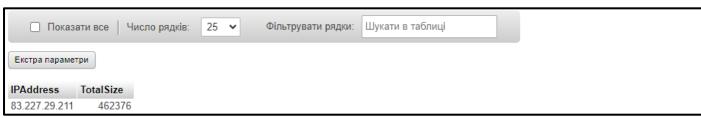
2.2. Придумайте будь який запит з використанням UNION:

```
1 SELECT
   SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' ', 1), ' ', -1) AS IPAddress,
      SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ']', -1), ' ', 5) AS LogRecord
   4 FROM
   5 tblaccesslog
   6 WHERE
       (Line LIKE '%GET%' AND Line LIKE '%HTTP/1.1\" 301%')
       OR (Line LIKE '%POST%' AND Line LIKE '%HTTP/1.1\" 404%')
   10 UNION ALL
   11
   12 SELECT
   13 ResponsibleLawyer,
   14 Note
   15 FROM
   16 responsibility
   17 WHERE
   18 Note IS NOT NULL
```



2.3. Знайдіть розмір всього скачаного комп'ютером з ІР адресою 83.227.29.211:





3. SQL-запити для створення всієї структури БД.

```
-- Вивести повні адреси JS скриптів (розширення js) і їх розмір, які запросили
-- з комп'ютера з IP адресою 83.227.29.211, для яких запит завершився вдало.
SELECT
  SUBSTRING_INDEX(SUBSTRING_INDEX(Line, '', 1), '', -1) AS IPAddress,
SUBSTRING_INDEX(SUBSTRING_INDEX(Line, '[', -1), '', 1) AS LogDate,
SUBSTRING_INDEX(SUBSTRING_INDEX(Line, '', 7), '', -1) AS FilePath,
SUBSTRING_INDEX(SUBSTRING_INDEX(Line, '', 9), '', -1) AS HttpStatusCode,
SUBSTRING_INDEX(SUBSTRING_INDEX(Line, '', 10), '', -1) AS ByteSize
FROM
  tblaccesslog
WHERE
  Line LIKE ('83.227.29.211%.js%')
   AND SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' ', 9), ' ', -1) BETWEEN '200' AND '206'
-- Придумайте будь який запит з використанням UNION
SELECT
  SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' ', 1), ' ', -1) AS IPAddress, SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ']', -1), ' ', 5) AS LogRecord
FROM tblaccesslog
WHERE
   (Line LIKE '%GET%' AND Line LIKE '%HTTP/1.1\" 301%')
   (Line LIKE '%POST%' AND Line LIKE '%HTTP/1.1\" 404%')
UNION ALL
SELECT
  ResponsibleLawyer,
  Note
FROM
  responsibility
WHERE
  Note IS NOT NULL
-- Знайдіть розмір всього скачаного комп'ютером з ІР адресою 83.227.29.211
  IPAddress, SUM(ByteSize) AS TotalSize
FROM (
   SELECT
     SUBSTRING_INDEX(SUBSTRING_INDEX(Line, '', 1), '', -1) AS IPAddress,
SUBSTRING_INDEX(SUBSTRING_INDEX(Line, 'GET ', -1), 'HTTP', 1) AS FilePath,
SUBSTRING_INDEX(SUBSTRING_INDEX(Line, '', 9), '', -1) AS HttpStatusCode,
SUBSTRING_INDEX(SUBSTRING_INDEX(Line, '', 10), '', -1) AS ByteSize
   FROM
      tblaccesslog
  WHERE Line REGEXP ('^83\.227\.29\.211*'))
AS DownloadedByIP
```

Контрольні питання

1. Що таке запити розширеного SQL (advanced SQL)?

Запити розширеного SQL – це високорівневі SQL-операції, які виходять за рамки базового синтаксису SQL. Вони включають у себе різноманітні функції, операції та можливості, які дозволяють більш гнучко та ефективно взаємодіяти з базами даних. До них можуть входити агрегатні функції, віконні функції, операції над множинами, аналітичні функції та інші.

2. Що таке регулярні вирази?

Регулярні вирази — це шаблони для пошуку та обробки тексту. Вони дозволяють визначати певні правила для пошуку, вилучення або заміни рядків в текстових данних. Регулярні вирази широко використовуються в програмуванні, редакторах тексту, операціях обробки даних та запитах до баз даних для більш гнучкого та потужного пошуку та аналізу текстової інформації.

3. Для чого використовуються запити з командою UNION?

Запити з командою **UNION** використовуються для об'єднання результатів двох або більше SELECTзапитів у єдиний результат. UNION видаляє дублікати, об'єднуючи унікальні рядки. Це дозволяє комбінувати дані з різних таблиць чи умов в один результат.

4. Що таке агрегатні функції?

Агрегатні функції – це функції в SQL, які виконують обчислення на підсумкових даних, таких як сума, середнє значення, мінімум, максимум тощо. Ці функції застосовуються до груп рядків та повертають одне значення для кожної групи.

5. Для чого використовуються запити з командою ORDER?

Запити з командою **ORDER** використовуються для сортування результатів запиту. ORDER BY дозволяє вказати порядок сортування для одного чи кількох стовпців у вибірці даних. Можна сортувати дані в порядку зростання (ASC) або спадання (DESC).

6. Для чого використовуються запити з командою HAVING?

Запити з командою **HAVING** використовуються для фільтрації результатів групованих даних на основі агрегатних функцій. HAVING використовується після GROUP BY і дозволяє вибирати групи, які відповідають певним умовам. Це дає можливість фільтрувати дані після групування та агрегації.

Висновки

У ході виконання лабораторної роботи було освоєно написання Advanced SQL-запитів.

Було використано можливості функції SUBSTRING_INDEX для вилучення та обробки інформації з лог-файлів. Команда UNION була необхідна для комбінування результатів різних запитів, що ϵ зручно для об'єднання даних із різних джерел чи умов. Також, використання агрегатних функцій (SUM) було необхідне для обчислення підсумкових значень та виведення загального обсягу завантажень для конкретної IP-адреси.