Каркушевський Владислав Леонідович

№ 16 Облік телефонних переговорів

Лабораторна робота №4

Вибірка даних з використанням розділів GROUP BY і HAVING.

|  |  |
| --- | --- |
| Завдання 1. А) використати агрегатні функції MAX (), MIN (), AVG () та COUNT() без GROUP BY | |
| Скрипт | Скрін |
| #без використання group by  SELECT  COUNT(\*) AS 'count',  AVG(Duration\_call) AS 'average number of calls',  MIN(Duration\_call) AS 'smallest call',  MAX(Duration\_call) AS 'biggest call',  ROUND(SUM(Cost), 4) AS'total cost'  FROM calls; |  |
| Завдання 1. Б) використати агрегатні функції MAX (), MIN (), AVG () та COUNT() із застосуванням GROUP BY; | |
| Скрипт | Скрін |
| #з використання group by  SELECT  ID\_abonent,  COUNT(\*) AS 'count',  AVG(Duration\_call) AS 'average number of calls',  MIN(Duration\_call) AS 'smallest call',  MAX(Duration\_call) AS 'biggest call',  ROUND(SUM(Cost), 4) AS 'total cost'  FROM calls  GROUP BY ID\_abonent; |  |
| Завдання 2 А) пояснити дію розділу GROUP BY на прикладах без використання агрегатних функцій; | |
| Скрипт | Результат |
| #без використання агрегатних функцій  SELECT  \*  FROM  calls  GROUP BY ID\_abonent; |  |
| Завдання 2 Б) пояснити дію розділу GROUP BY на прикладах з використанням агрегатних функцій; | |
| Скрипт | Результат |
| #з використання агрегатних функцій  SELECT  ID\_abonent,  ROUND(SUM(Cost), 4) AS'total cost'  FROM  calls  GROUP BY ID\_abonent; |  |
| Завдання 3. А) створити запит з використанням розділу HAVING для завдання умов, що містять агрегатні функції | |
| Скрипт | Результат |
| #з використання having  SELECT  ID\_abonent,  COUNT(\*) AS 'count',  AVG(Duration\_call) AS 'average number of calls',  MIN(Duration\_call) AS 'smallest call',  MAX(Duration\_call) AS 'biggest call',  ROUND(SUM(Cost), 4) AS 'total cost'  FROM  calls  GROUP BY 1  HAVING count >= 3  ORDER BY 2 DESC; |  |
| Завдання 3. Б) створити запит з використанням розділу WHERE для завдання умов, що містять агрегатні функції | |
| Скрипт | Результат |
| #з використання where  SELECT  ID\_abonent,  COUNT(\*) AS 'count',  AVG(Duration\_call) AS 'average number of calls',  MIN(Duration\_call) AS 'smallest call',  MAX(Duration\_call) AS 'biggest call',  ROUND(SUM(Cost), 4) AS 'total cost'  FROM calls  WHERE  Duration\_call BETWEEN 5 AND 25  GROUP BY ID\_abonent; |  |
| Завдання 3. В) створити запит з використанням розділу WHERE та HAVING для завдання умов, що містять агрегатні функції; | |
| Скрипт | Результат |
| #з використання where та having  SELECT  ID\_abonent,  COUNT(\*) AS 'count',  AVG(Duration\_call) AS 'average number of calls',  MIN(Duration\_call) AS 'smallest call',  MAX(Duration\_call) AS 'biggest call',  ROUND(SUM(Cost), 4) AS 'total cost'  FROM  calls  WHERE  Duration\_call BETWEEN 5 AND 25  GROUP BY ID\_abonent  HAVING count >= 3  ORDER BY 2 DESC  ; |  |
| Завдання 4 пояснити відмінність між розділами GROUP BY та ORDER BY, GROUP BY та DISTINCT. | |
| Результат | |
| ORDER BY використовується для впорядкування результатів запиту за певними полями, які можуть бути вказані в порядку зростання (ASC) або спадання (DESC).  GROUP BY використовується для групування даних за певними полями. Також можна використовувати агрегатні функції SUM, AVG, MAX тощо.  DISTINCT використовується для відображення унікальних значень у колонці таблиці. | |
| Приклад | |
| Наприклад, дано таблицю:    За допомогою ORDER BY отримаємо сортування:    За допомогою DISTINCT унікальні значення    А за допомогою GROUP BY можна порахувати кількість входжень | |