SQL Задачи и решения

Учебник. Сергей Моисеенко.

Накопительные итоги стр. 1

Коррелирующие подзапросы можно использовать для вычисления накопительных итогов - задачи, часто возникающей на практике.

В предположении некоторой упорядоченности строк накопительный итог для каждой строки представляет собой сумму значений некоторого числового столбца для этой строки и всех строк, расположенных выше данной.

Другими словами, накопительный итог для первой строки в упорядоченном наборе будет равен значению в этой строке. Для любой другой строки накопительный итог будет равен сумме значения в этой строке и накопительного итога в предыдущей строке.

Рассмотрим, например, такую задачу.

Для пункта 2 по таблице Outcome_o получить на каждый день суммарный расход за этот день и все предыдущие дни.

Вот запрос, который выводит информацию о расходах на пункте 2 в порядке возрастания даты

Выполнить Консоль

```
1.  SELECT point, date, out
2.  FROM Outcome_o o
3.  WHERE point = 2
4.  ORDER BY date;
```

point	date	out
2	2001-03-22 00:00:00.000	1440.00
2	2001-03-29 00:00:00.000	7848.00
2	2001-04-02 00:00:00.000	2040.00

Фактически, чтобы решить задачу нам нужно добавить еще один столбец, содержащий накопительный итог (run_tot). В соответствии с темой, этот столбец будет представлять собой коррелирующий подзапрос, в котором для ТОГО ЖЕ пункта, что и у ТЕКУЩЕЙ строки включающего запроса, и для всех дат, меньших либо равных дате ТЕКУЩЕЙ строки включающего запроса, будет подсчитываться сумма значений столбца out:

Выполнить Консоль

```
1.  SELECT point, date, out,
2.    (SELECT SUM(out)
3.     FROM Outcome_o
4.     WHERE point = o.point AND date <= o.date) run_tot
5.  FROM Outcome_o o
6.  WHERE point = 2
7.  ORDER BY point, date;
```

point	date	out	run_tot
2	2001-03-22 00:00:00.000	1440.00	1440.00
2	2001-03-29 00:00:00.000	7848.00	9288.00
2	2001-04-02 00:00:00.000	2040.00	11328.00

Собственно, использование пункта 2 продиктовано желанием уменьшить результирующую выборку. Чтобы получить накопительные итоги для каждого из пунктов, имеющихся в таблице Outcome_o, достаточно закомментировать строку

```
1.  WHERE point = 2
```

Ну а чтобы получить "сквозной" накопительный итог для всей таблицы нужно, видимо, убрать условие на равенство пунктов:

```
1.  point= o.point
```

Однако при этом мы получим один и тот же накопительный итог для разных пунктов, работавших в один и тот же день. Вот подобный фрагмент из результирующей выборки,

point	date	out	run_tot
1	2001-03-29 00:00:00.000	2004.00	33599.00
2	2001-03-29 00:00:00.000	7848.00	33599.00

Но это не проблема, если понять, что же мы хотим в итоге получить. Если нас интересует накопление расхода по дням, то нужно из выборки вообще исключить пункт и суммировать расходы по дням:

Выполнить Консоль

```
1.  SELECT date, SUM(out) out,
2.    (SELECT SUM(out)
3.     FROM Outcome_o
4.     WHERE date <= o.date) run_tot
5.  FROM Outcome_o o
6.  GROUP BY date
7.  ORDER BY date;
```

В противном случае, нам нужно указать порядок, в котором к накопительному итогу будут добавляться расходы пунктов в случае, когда у нескольких пунктов совпадает дата. Например, упорядочим их по возрастанию номеров:

Выполнить Консоль

```
1.  SELECT point, date, out,
2.    (SELECT SUM(out)
3.     FROM Outcome_o
4.     WHERE date < o.date OR ( date = o.date AND point <= o.point)) run_tot
5.  FROM Outcome_o o
6.  ORDER BY date, point;
```

Следующая страница

Рекомендуемые упражнения: **69**, **101**, **134**

коррелирующие подзапросы накопительные итоги

Предыдущая [Коррелирующие подзапросы] [Преобразование даты в строку] Следующая

Последние изменения:

Приложение 2. Список задач

Футбол

Тэги:

поиск по тэгам

ALL AND AUTO_INCREMENT AVG battles CASE CAST CHAR CHARINDEX CHECK classes COALESCE CONSTRAINT Convert COUNT CROSS APPLY CTE DATEADD DATEDIFF DATENAME DATEPART DATETIME DDL DEFAULT DELETE DISTINCT DML EXCEPT EXISTS EXTRACT FOREIGN KEY FROM FULL JOIN GROUP BY Guadalcanal HAVING IDENTITY IN INFORMATION_SCHEMA INNER JOIN insert INTERSECT IS NOT NULL IS NULL ISNULL laptop LEFT LEFT OUTER JOIN LEN maker

Больше тэгов

Учебник обновлялся
месяц назад
https://exchangesumo.com/obmen/to,