SQL Оконные функции. Часть 2

ГАЙДпо панатитний

СПИСОК ОКОННЫХ ФУНКЦИЙ:

Функции агрегации

- avg()
- · count()
- · max()
- · min()
- sum()

Функции ранжирования

- row number()
- rank()
- dense_rank()

Функции распределения

- percent_rank()
- cume_dist()

Аналитические функции

- ·lead()
- ·lag()
- ·ntile()
- irst_value()
- last_value()
- nth_value()

ФУНКЦИИ АГРЕГАЦИИ

- avg(expr) среднее значение в строках оконного фрейма
- count(expr) количество значений в строках оконного фрейма
- max(expr) максимальное значение в оконном фрейме
- min(expr) манимальное значение в оконном фрейме
- sum(expr) сумма значений в оконном фрейме

ORDER BY и оконный фрейм:

Функции агрегации не требуют ORDER BY. Они принимают определение оконного фрейма (ROWS, RANGE, GROUPS).

ФУНКЦИИ РАНЖИРОВАНИЯ

- row_number() уникальный номер для каждой строки в партиции (вне зависимости от связанности значений номера будут разными)
- rank() ранжирование в партиции, с пробелами и одинаковым ранжированием для связанных значений (после строк с одинаковым рангом будет номер строки)
- dense_rank() ранжирование в партиции, без пробелов с одинаковым ранжированием для связанных значений

city	price	row_number	rank	dense_rank		
city	price	over(order by price)				
Paris	7	1	1	1		
Rome	7	2	1	1		
London	8.5	3	3	2		
Berlin	8.5	4	3	2		
Moscow	9	5	5	3		
Madrid	10	6	6	4		
Oslo	10	7	6	4		

ORDER BY и оконный фрейм: rank() и dense_rank() требуют ORDER BY, но row_number() не требует ORDER BY. Функции ранжирование не принимают определение оконного фрейма (ROWS, RANGE, GROUPS).

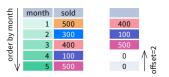
АНАЛИТИЧЕСКИЕ ФУНКЦИИ

- **lead**(*expr*, *offset*, *default*) возвращает значение, следующее за текущим с указанным шагом. *offset u default* опциональные; по умолчанию *offset* = 1, *default* = NULL
- lag(expr, offset, default) возвращает значение, предыдущее от текущего с указанным шагом. offset и default опциональные; по умолчанию offset = 1, default = NULL

lead(sold) OVER(ORDER BY month)

뒫	month	sold	
١٥	1	500	300
order by month	2	300	400
e l	3	400	100
2	4	100	500
-₩	5	500	NULL

lead(sold, 2, 0) OVER(ORDER BY month)



lag(sold) OVER(ORDER BY month)

뒫	month	sold	
ou	1	500	NULL
order by month	2	300	500
er	3	400	300
ord	4	100	400
Ĭ	5	500	100

lag(sold, 2, 0) OVER(ORDER BY month)

달	month	sold		. 7
order by month	1	500	0	Fset=2
충	2	300	0	\ ₩
e	3	400	500	Ī
P	4	100	300	
1	5	500	400	

 ntile(n) – делит строки в партиции на максимально равные n группы и присваивает каждой строке свой номер группы.



ORDER BY и оконный фрейм: ntile(), lead(), and lag() требуют ORDER BY. Они не принимают определение оконного фрейма (ROWS, RANGE, GROUPS).

ФУНКЦИИ РАСПРЕДЕЛЕНИЯ

- percent_rank() номер ранжирования строки в процентиле—значение в интервале [0, 1]: (ранг 1) / (общее кооличество строк 1)
- cume_dist() сводное распределение значения в группе значений, например номера строк
 со значениями меньше или равно значения текущей строки поделенной на общее количество
 строк; значение в инте рвале (0, 1]:

percent_rank() OVER(ORDER BY sold)

city	sold	percent_rank	
Paris	100	0	
Berlin	150	0.25	
Rome	200	0.5	<
Moscow	200	0.5	не включая эту строку
London	300	1	50% значений меньше, чем значение этой
			чем значение этой строки

cume_dist() OVER(ORDER BY sold)

city	sold	cume_dist	
Paris	100	0.2	
Berlin	150	0.4	
Rome	200	0.8	←
Moscow	200	0.8	80 % значений
London	300	1	меньше или равны значению этой
			строки

ORDER BY и Оконные функции: Функции распределения требуют ORDER BY. Они не принимают определение оконного фрейма (ROWS, RANGE, GROUPS).

- first_value(expr) значение первой строки в оконном фрейме
- last_value(expr) значение последней строки в оконном фрейме

first_value(sold) OVER
(PARTITION BY city ORDER BY month)

city	month	sold	first_value
Paris	1	500	500
Paris	2	300	500
Paris	3	400	500
Rome	2	200	200
Rome	3	300	200
Rome	4	500	200

last_value(sold) OVER
(PARTITION BY city ORDER BY month
RANGE BETWEEN UNBOUNDED PRECEDING
AND UNBOUNDED FOLLOWING)

city	month	sold	last_value
Paris	1	500	400
Paris	2	300	400
Paris	3	400	400
Rome	2	200	500
Rome	3	300	500
Rome	4	500	500

Заметка: Обычно, c last_value() хочется использовать конструкцию RANGE BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING. С дефолтным оконным фреймом ORDER BY, RANGE UNBOUNDED PRECEDING, last_value() возвращает значение текущей строки

 nth_value(expr, n) – значение n-й строки внутри оконного фрейма; п должен быть числом nth value(sold, 2) OVER (PARTITION BY city

ORDER BY month RANGE BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING)

city	month	sold	nth_value
Paris	1	500	300
Paris	2	300	300
Paris	3	400	300
Rome	2	200	300
Rome	3	300	300
Rome	4	500	300
Rome	5	300	300
London	1	100	NULL

ORDER BY и оконный фрейм:

first_value(), last_value(), and nth_value()не требуют ORDER BY. Они принимают определение оконного фрейма (ROWS, RANGE, GROUPS).