

SQL Window Functions Cheat Sheet

techfitflow.com | SQL Server syntax; Oracle and Snowflake accept the core | Test in non-production first

1. Skeleton

```
function(...) OVER (PARTITION BY group_cols ORDER BY sort_cols, unique_tiebreaker  
                    ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW)
```

2. Twelve patterns

#	Pattern	Function	Trap
1	Latest row per key	ROW_NUMBER in CTE, filter rn = 1	No tie-breaker = random row
2	Ties	RANK skips; DENSE_RANK does not	Top N may return more than N
3	Top N per group	DENSE_RANK over COUNT(*)	Decide if ties stay
4	Previous row	LAG(col,1,default)	First row is null
5	Running total	SUM OVER ... ROWS frame	Default RANGE shares ties
6	Moving average	AVG OVER ROWS 6 PRECEDING	ROWS counts rows, not days
7	Share of total	x / NULLIF(SUM(x) OVER (),0)	Filtered total
8	Period change	Aggregate, then LAG	Missing months
9	Gaps and islands	date - ROW_NUMBER	Dates must be unique
10	Buckets	NTILE, PERCENT_RANK, PERCENTILE_CONT	NTILE splits equal values
11	First/last value	FIRST_VALUE, LAST_VALUE	Extend frame for LAST_VALUE
12	De-duplicate	ROW_NUMBER, DELETE rn > 1	Review before delete

3. Which function?

- [] Compare to a neighbor row: LAG / LEAD; test the first row.
- [] Rank or pick rows per group: RANK or DENSE_RANK to keep ties; ROW_NUMBER for one row.
- [] Cumulative or moving value: SUM / AVG OVER with a ROWS frame.
- [] Share of a whole: SUM(x) OVER (PARTITION BY ...).
- [] Every branch: end ORDER BY with a unique column.

4. Platform differences

Behavior	SQL Server	Oracle	Snowflake
Nulls in ASC order	First	Last	Last (default)
NULLS FIRST/LAST	No (use CASE)	Yes	Yes
Filter on window result	CTE	Subquery	QUALIFY
Default frame with ORDER BY	RANGE	RANGE	RANGE
Delete through CTE	Yes	No	No

5. Core SQL

```
-- Latest per patient, deterministic
WITH r AS (SELECT PAT_ID, PAT_ENC_CSN_ID,
  ROW_NUMBER() OVER (PARTITION BY PAT_ID ORDER BY HOSP_ADMSN_TIME DESC, PAT_ENC_CSN_ID DESC) AS rn
  FROM PAT_ENC_HSP)
SELECT PAT_ID, PAT_ENC_CSN_ID FROM r WHERE rn = 1;

-- Running balance, explicit frame
SUM(amount) OVER (PARTITION BY account_id ORDER BY posting_date, posting_id
  ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW)

-- Days since prior discharge
DATEDIFF(day, LAG(HOSP_DISCH_TIME) OVER (PARTITION BY PAT_ID
  ORDER BY HOSP_ADMSN_TIME, PAT_ENC_CSN_ID), HOSP_ADMSN_TIME)
```

6. Specification template (BA)

Field	Your entry
Business question	
Output grain (one row per what)	
Partition key	
Order key and direction	
Tie-breaker	
Frame (rows or range, size)	
Null and first-row rule	
Three hand-checked sample rows	
Sign-off (name, date)	

7. QA test list

Test	Expected	Pass
Two rows with the same order key	Same output on every run	
Partition with one row	No error; LAG null or default	
Null order key	Placed where spec says	
Missing dates in daily series	Calendar fill or documented	
Intermediate rows of a running total	Each row correct, not only the last	
Run twice, compare with EXCEPT	Zero differing rows	

Sources: Microsoft Learn OVER clause; Snowflake window function reference; Oracle SQL Language Reference; CMS readmission program documentation. Keep results inside the approved environment; apply minimum necessary.