# **Using Statistical Functions**



Kathi Kellenberger

@auntkathi | auntkathisql.com

### Statistical Functions

# Top 10

- PERCENT\_RANK Calculate the rank over the group
- CUME\_DIST Calculate the relative position
- PERCENTILE\_CONT Given a rank, interpolate the value
- PERCENTILE\_DISC Given a rank, find the value

#### SYNTAX

```
PERCENT_RANK | CUME_DIST()
OVER([PARTITION BY <expression>] ORDER BY <expression>)
Formula
PERCENT_RANK: (RANK - 1)/(N - 1)
CUME_DIST: RANK/N
```

### Statistical Functions

# Top 10

- PERCENT\_RANK Calculate the rank over the group
- CUME\_DIST Calculate the relative position
- PERCENTILE\_CONT Given a rank, interpolate the value
- PERCENTILE\_DISC Given a rank, find the value

#### SYNTAX

```
PERCENTILE CONT | PERCENTILE DISC(<rank>)
WITHIN GROUP(ORDER BY <expression>)
OVER([PARTITION BY <expression>])
Median:
PERCENTILE_CONT(0.5)...
PERCENTILE_DISC(0.5)...
```

# Real World Examples

**Grading Curve** 

**Capacity Planning** 

### Performance



No frame to consider
Index helpful
Worktables in tempdb

# Summary



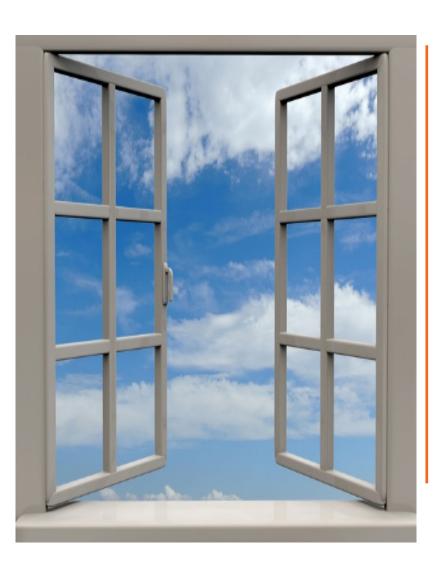
#### Functions added in 2005

- Ranking functions
- Window aggregates

#### Functions added in 2012

- Accumulating window aggregates
- Offset functions
- Statistical functions

# Summary



Makes writing queries easier

Functions perform over a set of rows

The OVER clause defines the window

- Partition by
- Order by
- Frame

# Summary



Covering index

Create worktables on disk

Window aggregates
Statistical functions

Specify frame with ROWS

- Accumulating window aggregates
- FIRST\_VALUE and LAST\_VALUE

Test, test, test