



Module 2 Day 2

Aggregate Functions / Subqueries

Order By

- Sorts rows by the named column(s)
- `SELECT col1, col2`
`FROM table`
`WHERE ...`
`ORDER BY col1 [ASC | DESC], col3 [ASC | DESC]...`
- Order-by column does not need to be in the select list
- Multiple Order-by columns “break ties”
- `ORDER BY n [ASC | DESC]`
 - Uses the nth column in the select list

Top N

- Limits the result set to the first N rows
- `SELECT TOP n col1, col2`
- “Top 10 most populous cities”

Functions

- Use wherever you'd use a column name
- String functions
 - Concatenation with +
 - CONCAT, LEFT/RIGHT, LEN, LOWER, UPPER, LTRIM, RTRIM, TRIM, STR, SUBSTR
- https://www.w3schools.com/sql/sql_ref_sqlserver.asp
 - Also Numeric and Date functions
- ISNULL(expression, valueIfNull)



Let's
Code

Aggregate Functions

- Performs some calculation on multiple rows and summarizes result
- `AVG`, `COUNT`, `MAX`, `MIN`, `SUM`
- `SELECT AVG(col1), MAX(col2)`
- `SELECT COUNT(*) FROM employee`
- This will always return one row!
- Column name will be undefined in the result set
 - You can name the column with `AS`

Let's
Code

Grouping Results

- Always used with aggregate functions
- Returns 1 row per “group”
- `SELECT col1, col2, FUNC1(col3)`
`FROM tablename`
`WHERE search-condition -- optional`
`GROUP BY col1, col2`
`ORDER BY ... -- optional`
- Columns NOT aggregated must be listed in the GROUP BY

Subquery

- One way of relating data between tables (or between a table and itself)
- `SELECT col1, col2...`
`FROM table1`
`WHERE col3 IN | NOT IN`
`(SELECT col9 FROM table2 WHERE ...)`
- Compares one column in the “outer” query to one column in the “inner” query