Let’s Tackle SQL

About Me

- Self-taught SQL programmer working with SQL Server 10+ years
- Full-stack web developer
- Enjoy learning new languages and techniques
- Runner and musician
- Contractor presently with Mayo Clinic

Technical Stack For This Talk

- **OS**: Windows 10
- **RDBMS (Relational Database Management System)**: SQL Server 2012
- **Application**: SQL Server Management Studio
- **Database**: AdventureWorks
- **Language**: T-SQL (Transact-SQL)

There are many other RDBMS', language versions and datasets that can be used for practice purposes on both Windows and macOS.

Versions of SQL vary slightly - once you understand the basics, you will be able to target the differences, just as you would with any other language.

Thinking About The Data You Need

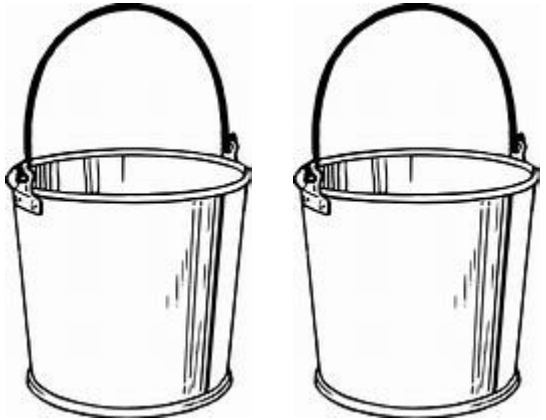
Questions

- What kind of data am I looking for?
- Do I know what tables this data is in?
- Do I need all the attributes or only a portion?

Tips

- Imagine tables as containers you can take things out of
- Think about how pieces of data may match together
- What questions are you trying to answer?

The Materials: FROM



- One of the 2 pieces you need to create a simple query
- Choose a specific table to review
- Can contain JOINs to link tables together
- Aliasing your tables for more complex queries

The Swiss Army Knife: SELECT

- One of the 2 pieces you need to create a simple query
- Allows you to peek inside tables and take a look around
- Return a portion of records for review

The Saw: WHERE

- Refine your simple query with conditions
- Use parentheses to ensure logic is evaluated correctly

The Screwdriver: JOINs

- Brings data together from multiple tables
- Primary/foreign key relationships make it easier to determine which columns to use
- INNER JOIN is the most common => matches data between all the tables in the join

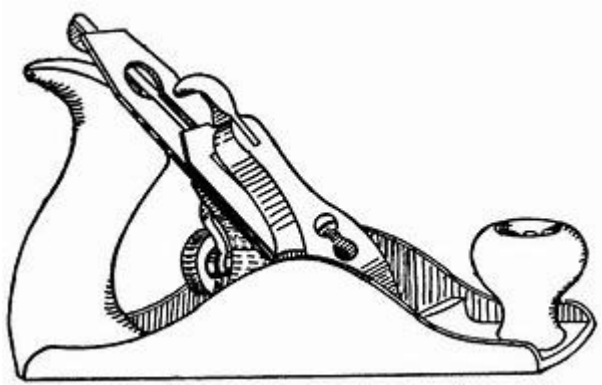
The Screwdriver: JOINs

- LEFT (OUTER) JOIN => all of the data from the “left” table, only matching data from the “right” table
- RIGHT (OUTER) JOIN => all of the data from the “right” table, only matching data from the “left” table
- FULL (OUTER) JOIN => brings back all rows from both tables, filling in missing values with NULLs
- There are selected cases where you would not use a JOIN but beware of Cartesian products!

The Wrench: GROUP BY

- Allows aggregation of data within the query
- Based on the selected fields, changes the granularity of the returned data
- Results can be further refined with the HAVING clause, which allows conditional logic on the grouped data

The Planer: ORDER BY



- Sorts the returned results
- Useful if the results will be further processed as ordered values
- Don’t use if you are able to use the dataset unsorted - there is extra processing power used that can cause queries to run longer than needed

The Clamp: UNION

- Allows combinations of returned result sets to be viewed as a single set of results
- Variations
  - UNION => returns distinct rows between the 2 sets
  - UNION ALL (recommended) => returns sets with no further operations taken
  - Other advanced variations outside the scope of this talk

Resources

Transact-SQL Documentation (SQL Server 2017):

https://docs.microsoft.com/en-us/sql/t-sql/language-reference?view=sql-server-2017

The Guru's Guide to Transact-SQL:

https://www.amazon.com/Gurus-Guide-Transact-SQL-Ken-Henderson/dp/0201615762/

SQL Shack's List of Top SQL Server Blogs:

https://www.sqlshack.com/sql-server-blogs/

(I recommend SQL Authority, MSSQLTips, and Simple Talk (RedGate))

Any friendly DBA or SQL guru - contact me anytime

Kristen Kinnear-Ohlmann

kristenkinnearohlmann

@kinnear_ohlmann

kristenkinnearohlmann

Questions?