SQL

Beginners are Awesome

Thank you for attending our 2024 conference!
Please do the following:

1) Log in to the conference website (easyregpro) by scanning the qr code.

2) Enter your confirmation number (found on your badge or in email)

3) Click on “My Schedule”. If you are not scheduled for this class, please “modify schedule”. This will allow you to provide feedback for the presenter and give us a count of attendees.

WIFI Password : GSIS2024

Ward Weatherford

30+ Years Technology Experience

- Product Development
- Software Engineering
- Project Management
- Customization
- Integration
- Database Design

Northwest RESA

What is RESA

- It’s Not Candy
- **R**egional **E**ducational **S**ervice **A**gency
- Every School District in GA is a member
- 16 RESAs
- Support 180 School Districts

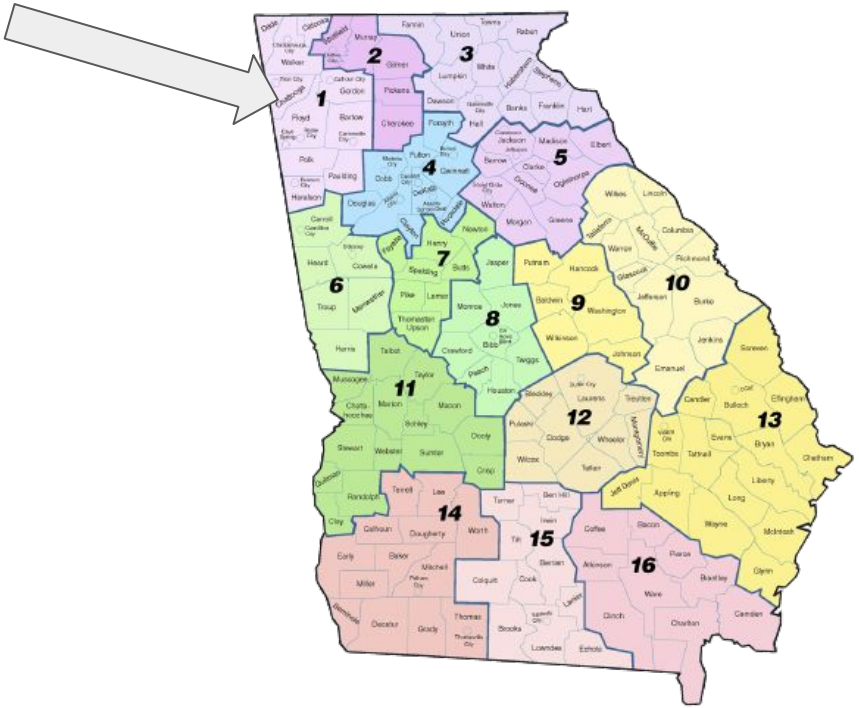
Required to provide

these service areas

- School Improvement
- Curriculum and Instruction
- Research and Planning
- Assessment and Evaluation
- Technology
- Health
- Professional Development

Northwest RESA

- 17 Member School Districts
- 25 Employees
- + GNETS Staffing



Northwest RESA

Whatsit

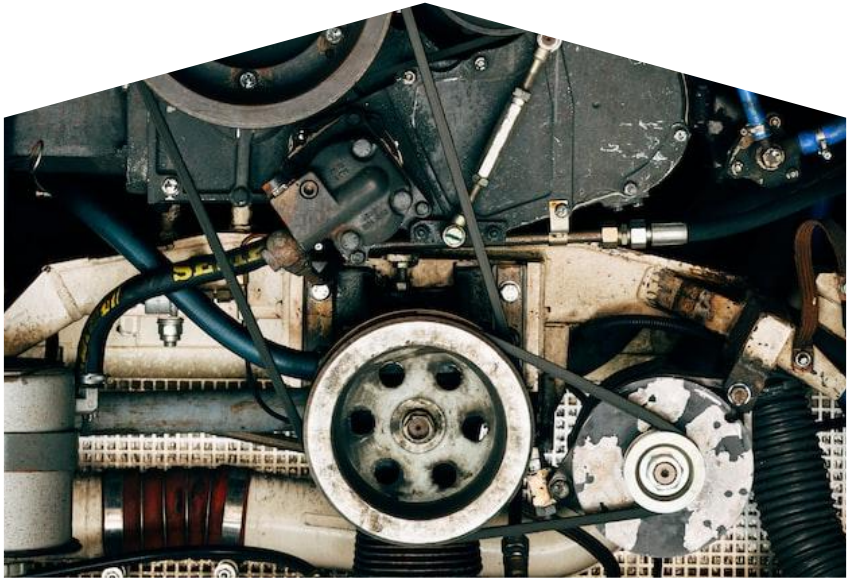


Photo by Michal Matlon on Unsplash

Northwest RESA

Brief History

- Relational Database Management System (RDBMS)
- Normalization
- **S**tructured **E**nglish **QU**ery **L**anguage
  - SEQUEL -> SQL
- Standardized in 1986 (ANSI) 1987 (ISO)
  - 1989, 1992, 1996, 1999, **2003, 2006, 2008, 2011, 2016, 2019**
- Vendor Specifics
  - Procedural Extension, Distributed Processing, Interoperability, Proprietary
- Nerd-Time
  - SQL is product of tuple relational and domain relational calculus.
  - Thank you Mr. Codd, Mr. Lacroix, and Mr. Pirotte

Normalization

Normalized

- Organize Data According to a List of Normal Forms
  - Thank You, Raymond Boyce
  - Six Normal Forms
  - Not All RDBMS Conform
  - Database Design Dependent
- Reduces Data Redundancy
- Improves Integrity

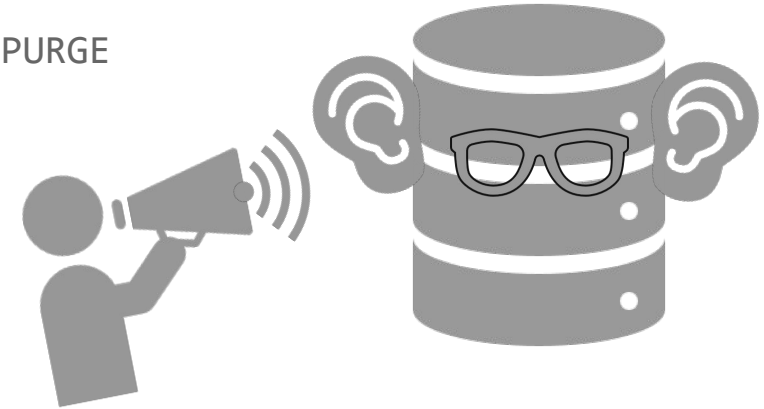
Denormalized

- Flatten Data
- Performance Improvements
- Report Centric


That's Just
The Way It Is

Types of SQL Statements

- Data Retrieval
  - `SELECT`
- Data Manipulation Language
  - `INSERT, UPDATE, DELETE, MERGE, LOCK, CALL`
- Data Definition Language
  - `ALTER, CREATE, DROP, RENAME, TRUNCATE, PURGE`
- Data Control Language
  - `GRANT, REVOKE`
- Transaction Control Language
  - `COMMIT, ROLLBACK, SAVEPOINT`



What Answer Are You Looking For?

- Load Up the Wayback Machine
  - Writing Composition Strategies
  - Who, What, Where, When, Why, How
- Determine the Target
- Get that Data Dictionary

Anatomy of a SELECT Statement

- `SELECT`
  - Get it all with a splat (*)
  - List fields
  - Expressions
- `FROM`
  - Tables
  - Views
  - CTE
- Optional
  - `WHERE` Clause
    - Filters Results
  - `ORDER BY` Clause
    - You guessed it. Orders the results
  - `GROUP BY` Clause
    - Provides grouping for aggregate expressions like SUM, AVG…
  - `HAVING` Clause
    - Filters groups

Operators

Operator	Condition	Example
=	Exact comparison	col_name = "value"
!= or <>	Exact inequality comparison	col_name != "value"
LIKE	Pattern search comparison % Matches a zero or more characters _ Matches a single character	col_name LIKE "value" col_name LIKE "%AN%" (matches "AN", "ANTIC", "TAN" or "ANTS") col_name LIKE "AN_" (matches "ANT", but not "AN")
BETWEEN	Ranged search	col_name BETWEEN 10 and 100
IN (...)	Exact match within a list of values	col_name IN ("Value1", "Value2", "Value3")
AND	Combines two conditions requiring both to be true	col_name1 = "value" AND col_name2 = "value"
OR	Combines two conditions at least one has to be true	col_name1 = "value" OR col_name2 = "value"
NOT	Negates the condition	col_name NOT IN(1, 2, 5) col_name NOT LIKE ("AN_") NOT (col_name1 = "value" AND col_name2 = "value")

Simple SQL Statement

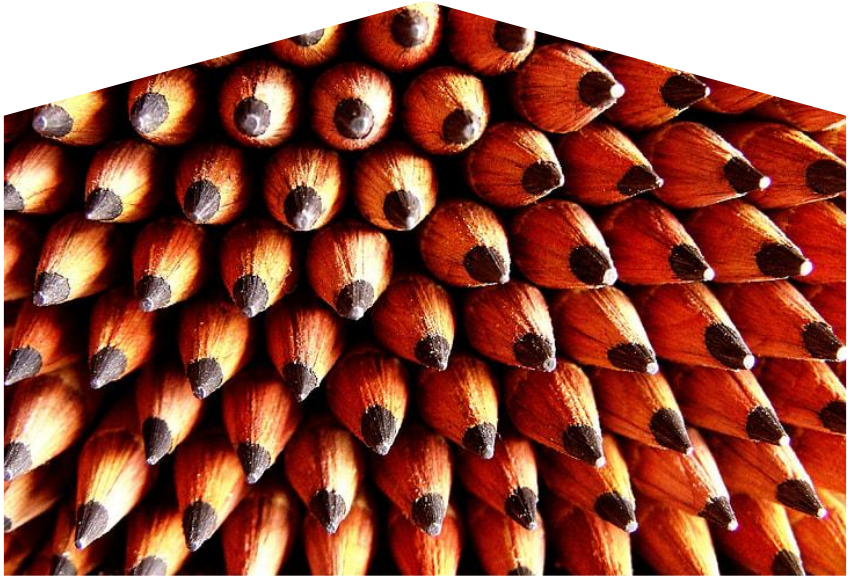
```
SELECT column1, column2 FROM table_name;

SELECT * FROM table_name
```

Combining Constraints

Use parentheses to make complex constraints

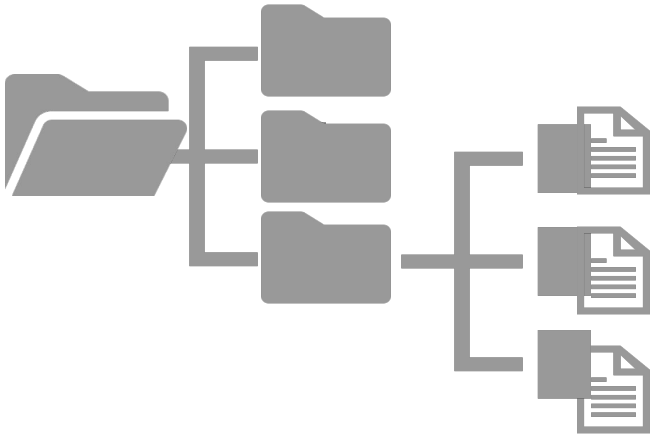
```
SELECT column1, column2
FROM table_name
WHERE column2 = "VALUE"
   AND ( column1 LIKE "AN_" OR column1 LIKE "_AT" )
```

TIPS



Organize Your Work

- Save Your Work
- Build a Folder System
- Text Search the Folders



PowerShell

```
Get-ChildItem *.sql -Recurse | Select-String "login" -List | Select Path
```

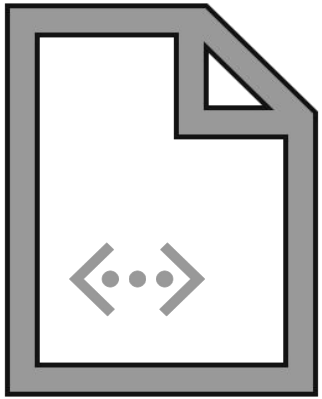
Bash

```
grep -r "login" *.sql | awk -F: '{print $1}'
```

Comments

Comment Your Queries

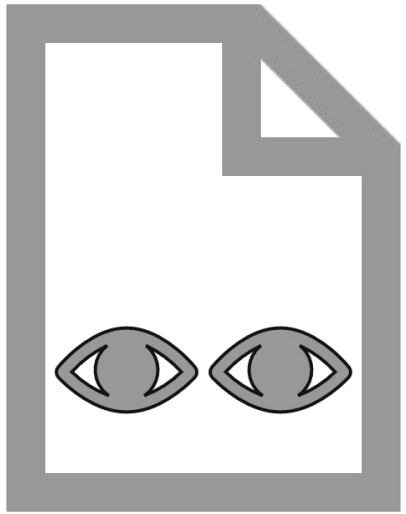
- Helps Others
- Complicated Code
- Poorly Named Fields
- Undocumented/Poorly Documented Schema



```
/* This is a block comment. Everything between the
slash and splat isn't processed by the interpreter */
select column1 -- Everything after the double dash is ignored
from table_name
```

Format for Readability

- What You Wrote Yesterday
  - Water Under the Bridge
- Capitalization
  - Functions
  - Keywords
  - Comments
- Consistency
- Helps Reuse



Resources

- Search Engines
- Data Dictionary
- https://www.w3schools.com/sql
- https://livesql.oracle.com
- https://sqlfiddle.com
- https://stackoverflow.com
- https://dbeaver.io