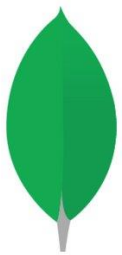
Relational Databases (SQL)

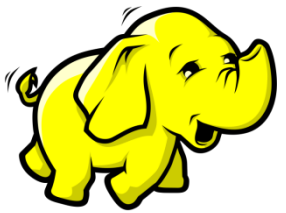
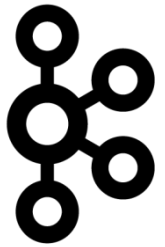
Michael Enudi

Journey through the world of databases and data engineering





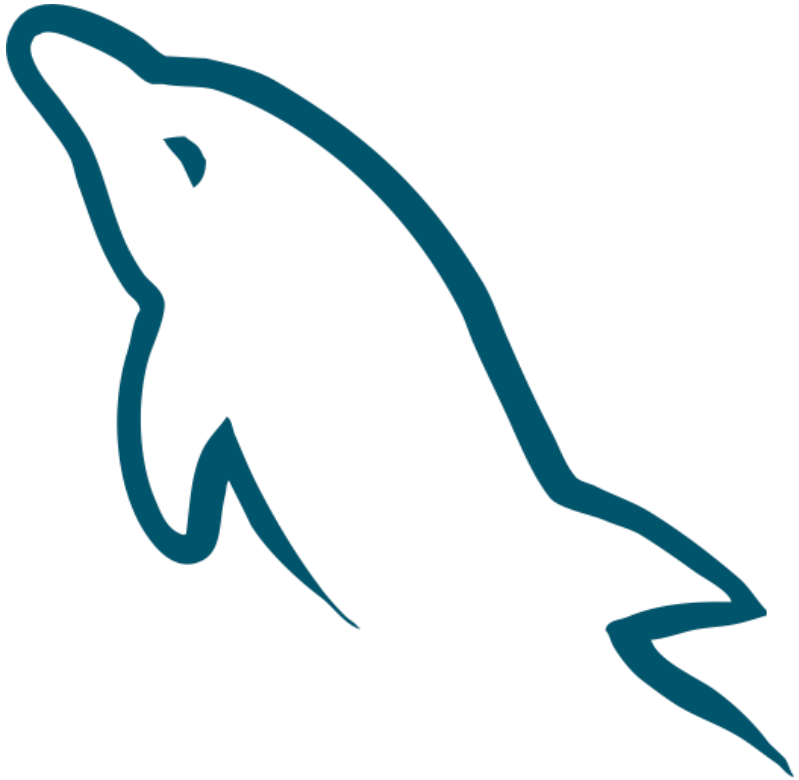


Note

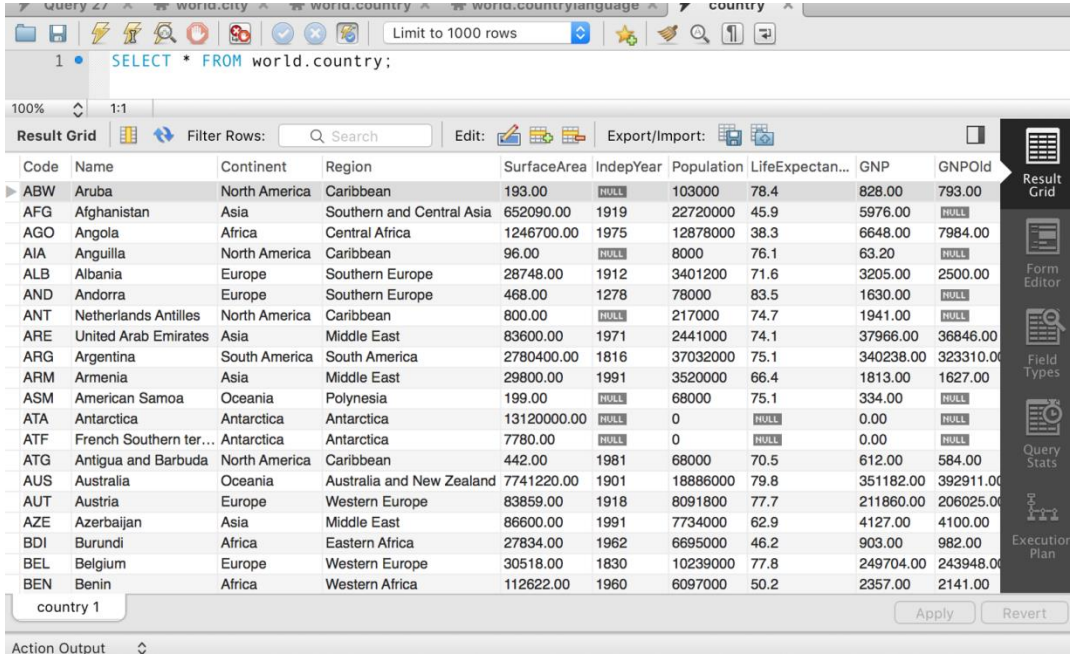
Scope

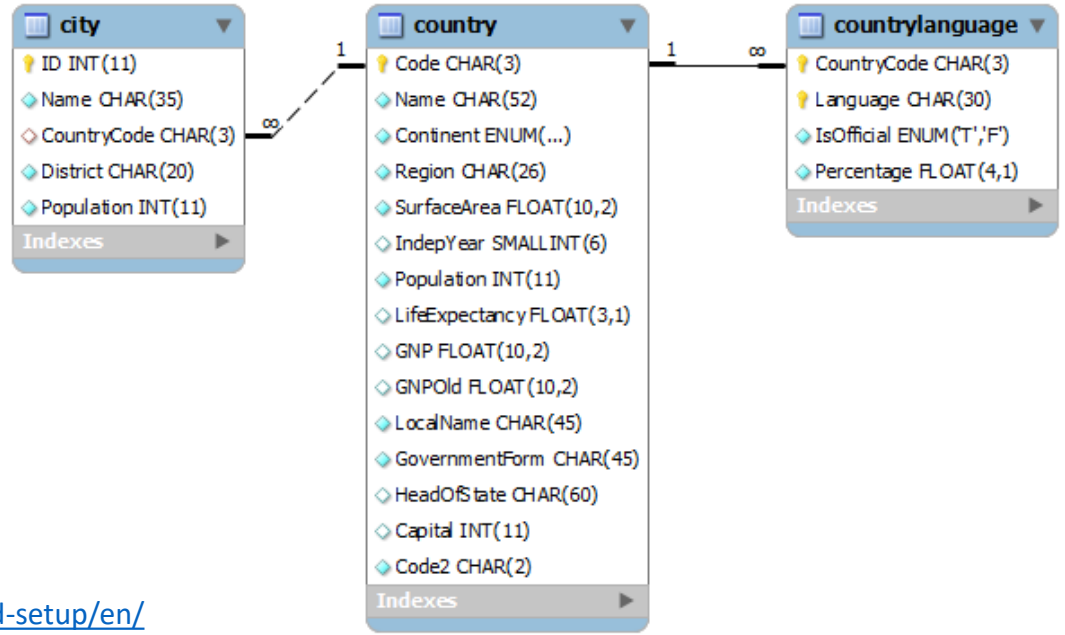
- Introduction to Relational Databases
- SQL
- MySQL
- Movielens in MySQL
- Data Processing
- Indexes
- OLTP Systems
- Data Warehousing
- Analytical Processing (OLAP)
- Database Logs
- Relational Databases - To be or Not to Be



Relational Databases

- Relations / Tables
- Record / Row / Tuple
- Primary Key(s)
- Fields / Columns / Properties
- Data types
- NULL
- Constraints
- Uniqueness
- Relationships
- Transactions (ACID)
- Index
- ERD
- RDBMS

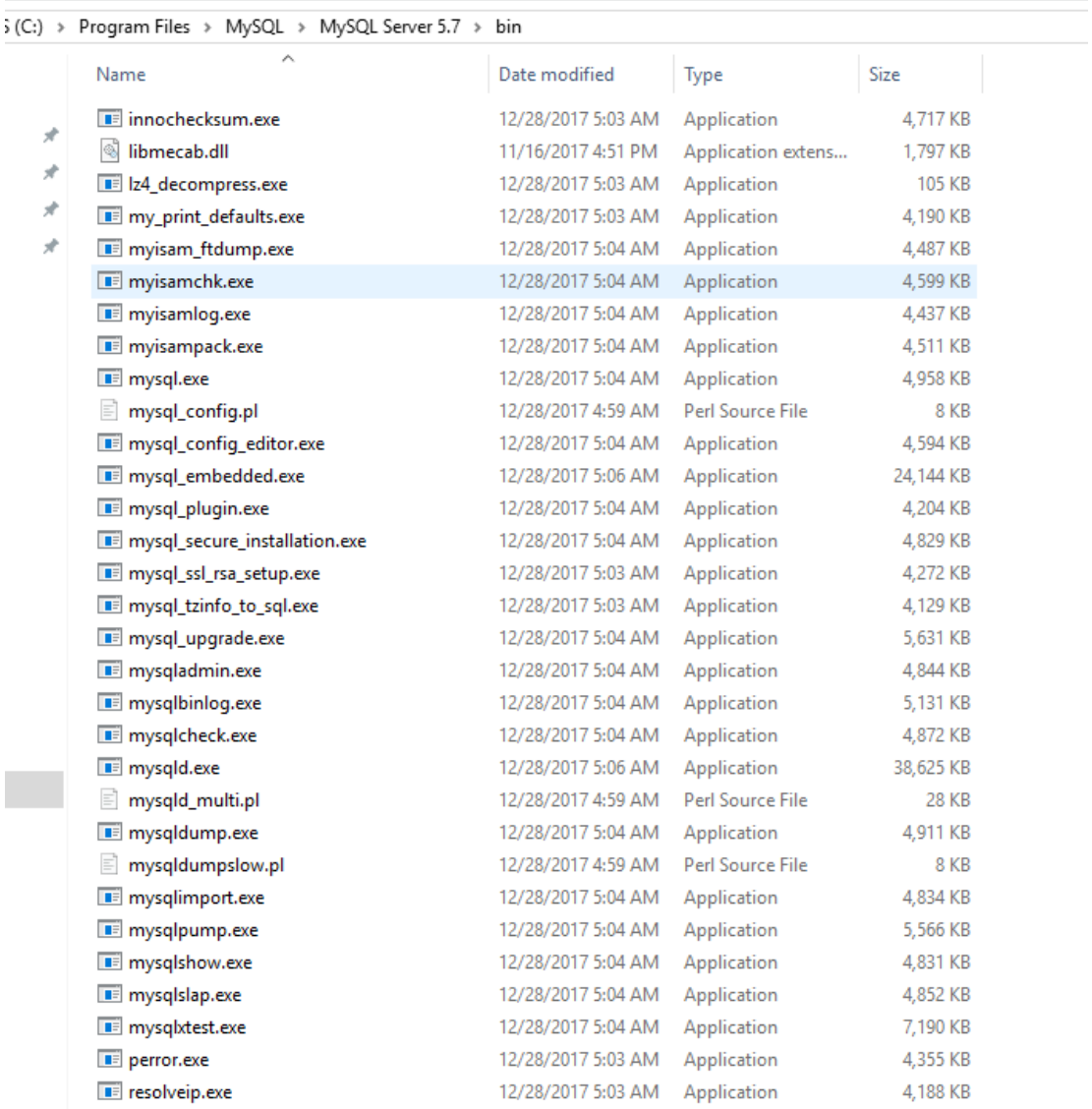




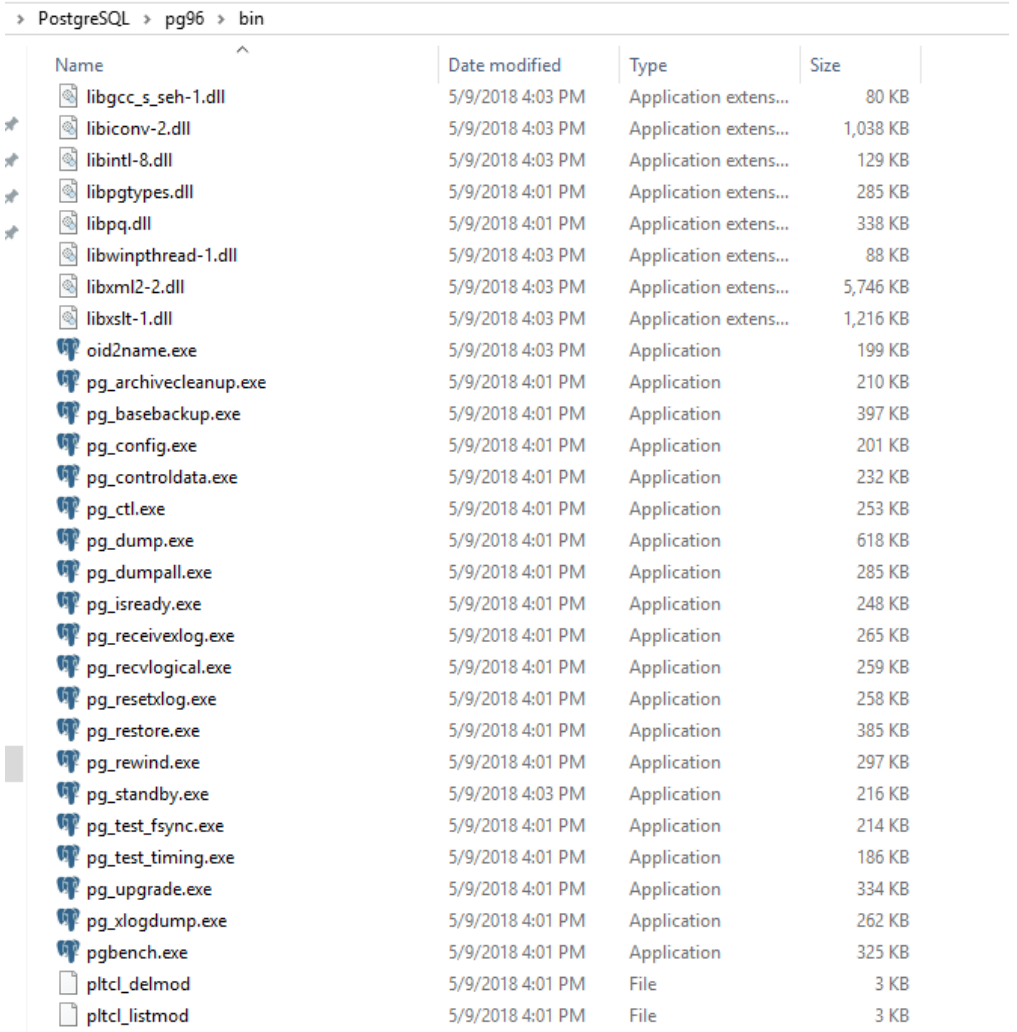
World database : https://dev.mysql.com/doc/world-setup/en/

RDBMS

MySQL



PostgresSQL



SQL

Article Talk Read Edit View history Search Wikipedia

Relational algebra

From Wikipedia, the free encyclopedia

Not to be confused with Relation algebra.

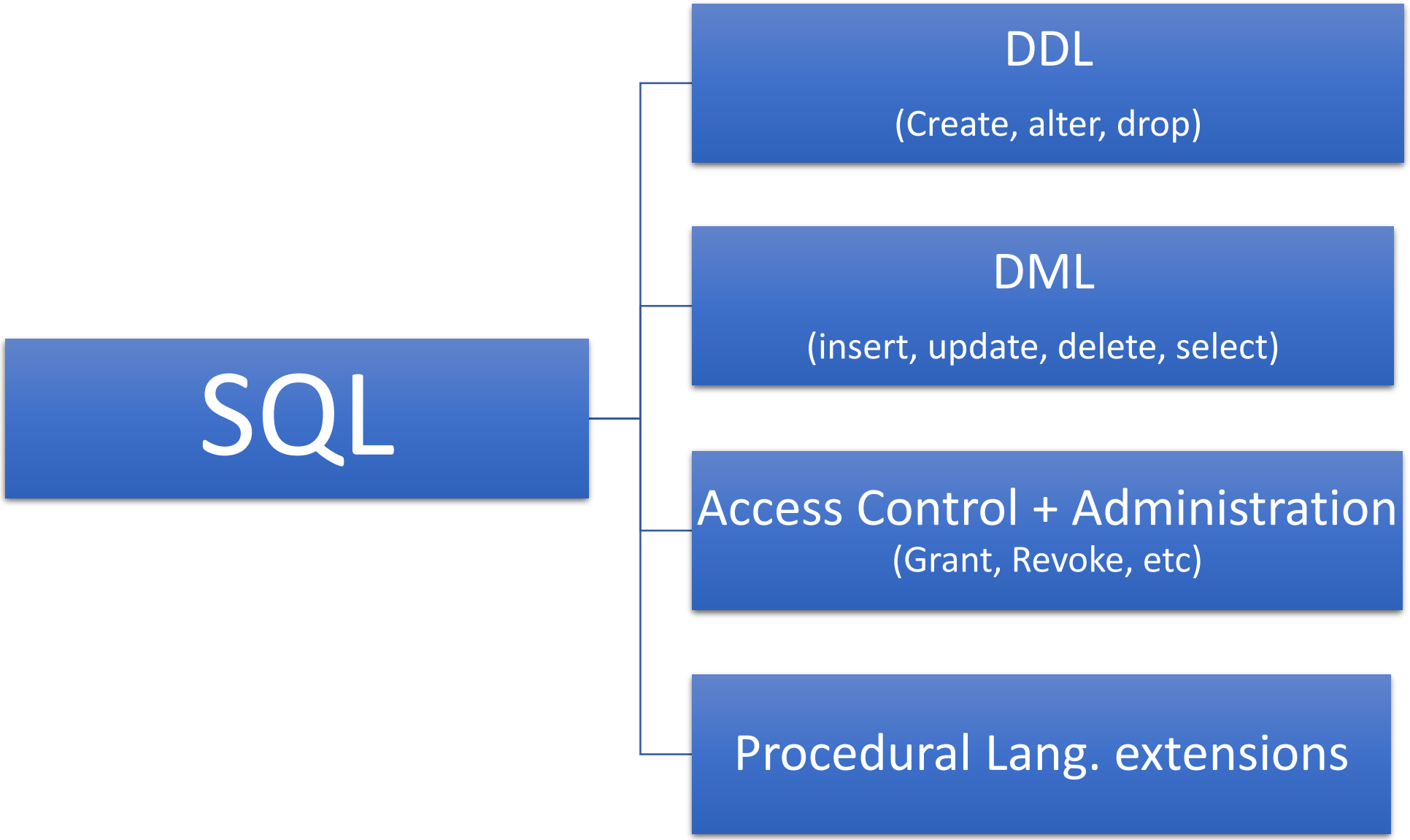
This article's **lead section does not adequately summarize key points of its contents**. Please consider expanding the lead to provide an accessible overview of all important aspects of the article. Please discuss this issue on the article's talk page. *(April 2015)*

Relational algebra, first created by Edgar F. Codd while at IBM, is a family of algebras with a well-founded semantics used for modelling the data stored in relational databases, and defining queries on it.

The main application of relational algebra is providing a theoretical foundation for relational databases, particularly query languages for such databases, chief among which is SQL.

Contents [hide]

SQL

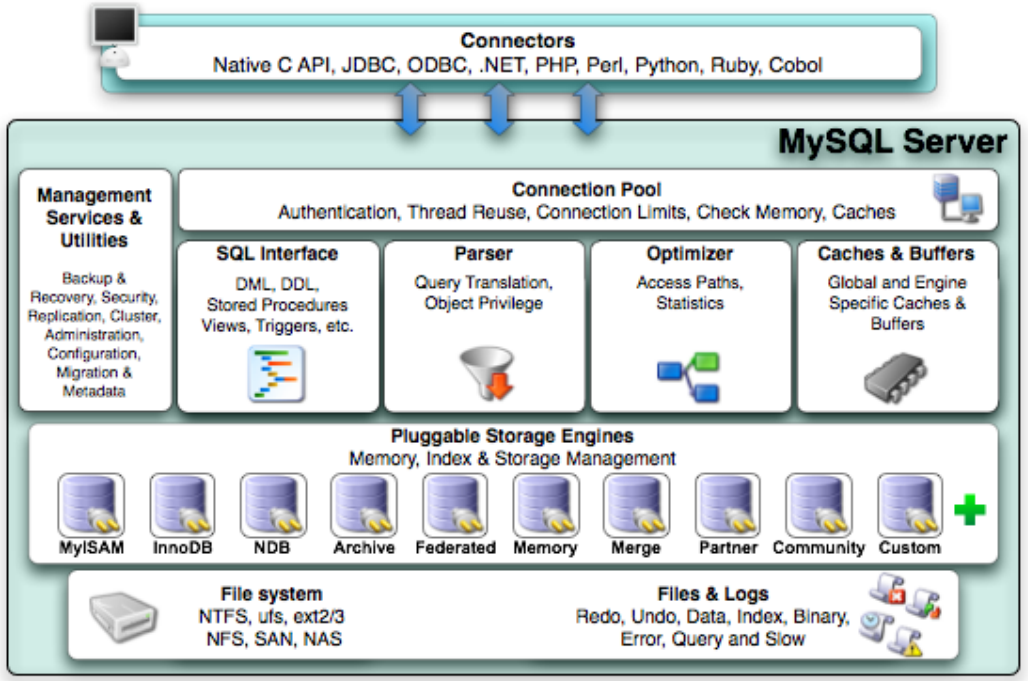

SQL
DDL
(Create, alter, drop)
DML
(insert, update, delete, select)
Access Control + Administration
(Grant, Revoke, etc)
Procedural Lang. extensions

SQL History

Year	Name	Alias	Comments
1986	SQL-86	SQL-87	First formalized by ANSI.
1989	SQL-89	FIPS 127-1	Minor revision that added integrity constraints, adopted as FIPS 127-1.
1992	SQL-92	SQL2, FIPS 127-2	Major revision (ISO 9075), *Entry Level* SQL-92 adopted as FIPS 127-2.
1999	SQL:1999	SQL3	Added regular expression matching, recursive queries (e.g. transitive closure), triggers, support for procedural and control-of-flow statements, non-scalar types (arrays), and some object-oriented features (e.g. structured types). Support for embedding SQL in Java (SQL/OLB) and vice versa (SQL/JRT).
2003	SQL:2003		Introduced XML-related features (SQL/XML), *window functions*, standardized sequences, and columns with auto-generated values (including identity-columns).
2006	SQL:2006		ISO/IEC 9075-14:2006 defines ways that SQL can be used with XML. It defines ways of importing and storing XML data in an SQL database, manipulating it within the database, and publishing both XML and conventional SQL-data in XML form. In addition, it lets applications integrate queries into their SQL code with XQuery, the XML Query Language published by the World Wide Web Consortium (W3C), to concurrently access ordinary SQL-data and XML documents.[34]
2008	SQL:2008		Legalizes ORDER BY outside cursor definitions. Adds INSTEAD OF triggers, TRUNCATE statement,[35] FETCH clause.
2011	SQL:2011		Adds temporal data (PERIOD FOR)[36] (more information at: Temporal database#History). Enhancements for *window functions* and FETCH clause.[37]
2016	SQL:2016		Adds row pattern matching, polymorphic table functions, JSON.

Source : https://en.wikipedia.org/wiki/SQL

MySQL



- Authored in 1995 by a Swedish company MySQL AB
- Currently developed by Oracle Incorporated
- Current version – 8.0.15 as at Feb. 2019
- According to db-engines
  - The second most used database in the world
  - The most deployed database in the world for web servers
- Written C and C++
- Vast language support
- Facebook has one of the known largest deployment of MySQL with over 1.3 billion users.

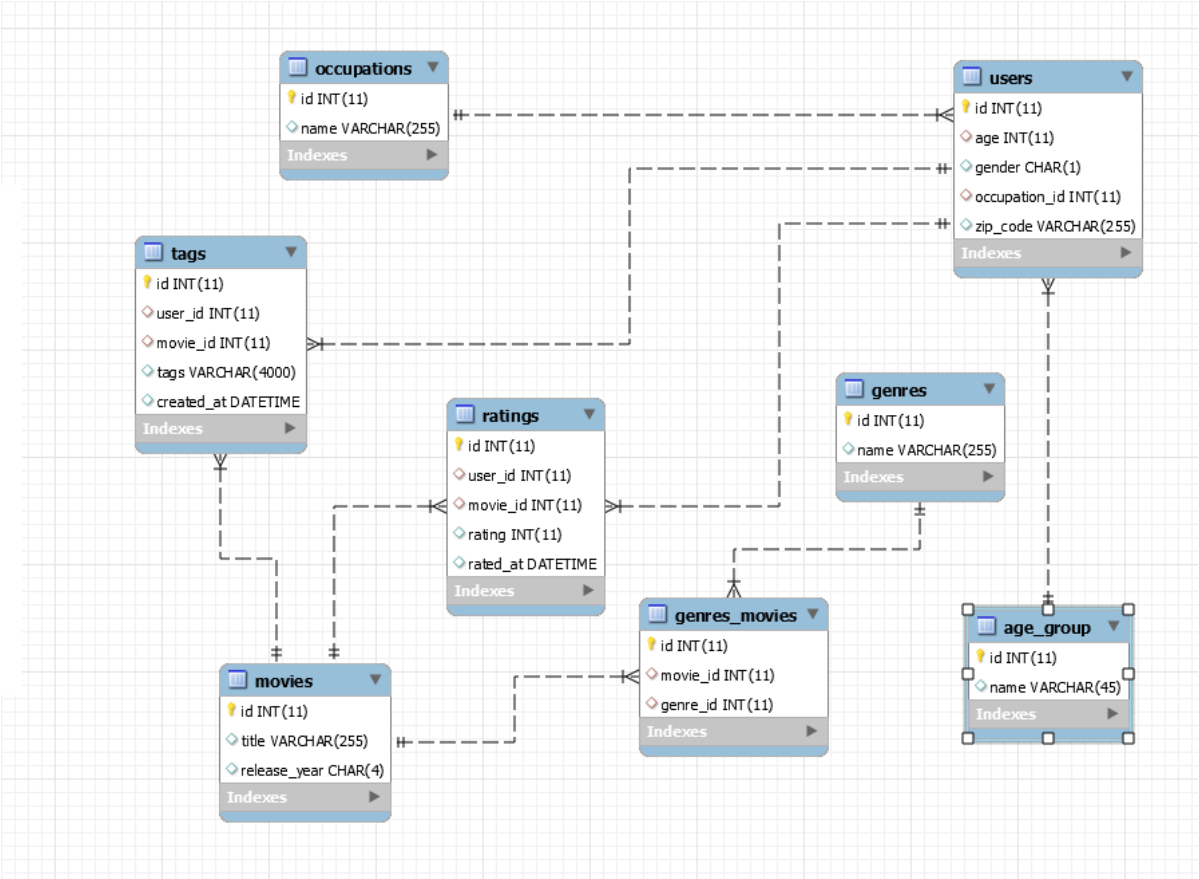
https://www.itworld.com/article/2831999/facebook--other-web-giants-unite-to-scale-mysql.html

- Pluggable and flexible storage engine architecture. Supports a number out of the box
- Default storage engine today is InnoDB

Movielens Dataset

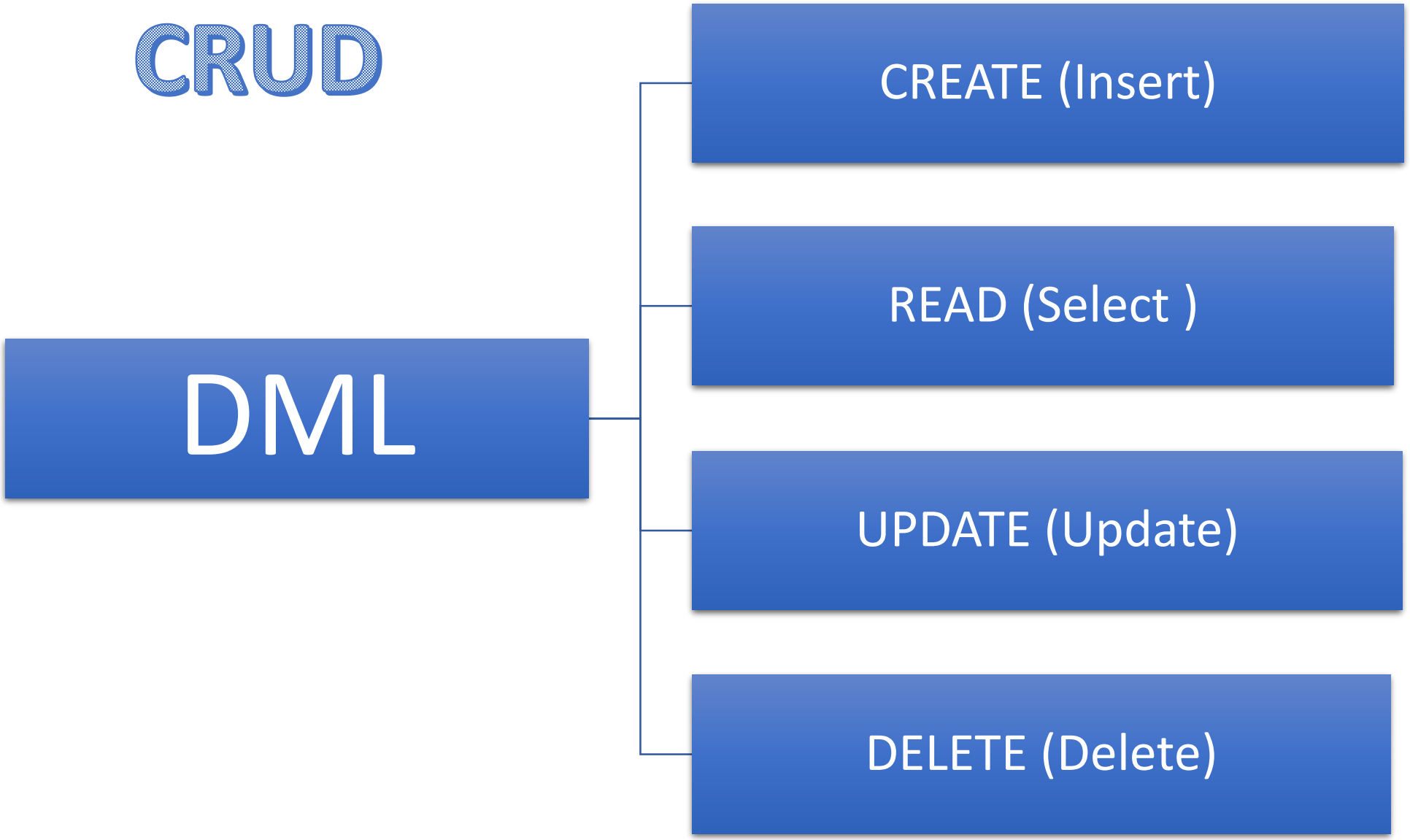
movielens

MovieLens is a web site that helps people find movies to watch. It has hundreds of thousands of registered users. We conduct online field experiments in MovieLens in the areas of automated content recommendation, recommendation interfaces, tagging-based recommenders and interfaces, member-maintained databases, and intelligent user interface design.



Movielens dataset : https://grouplens.org/datasets/movielens/ from GroupLens Research

Data Processing in RDBMS



Data Processing in RDBMS

In the application space

Example is LAMP stack or Django

Application data structures

Data logic is implemented in a programming language

Prevalent in multi-tiered architecture

Suited for OLTP kind of application where the amount of data transferred is relatively small.

In the database space

Example Microsoft Access Forms or Oracle Apex or Oracle Forms

Database objects

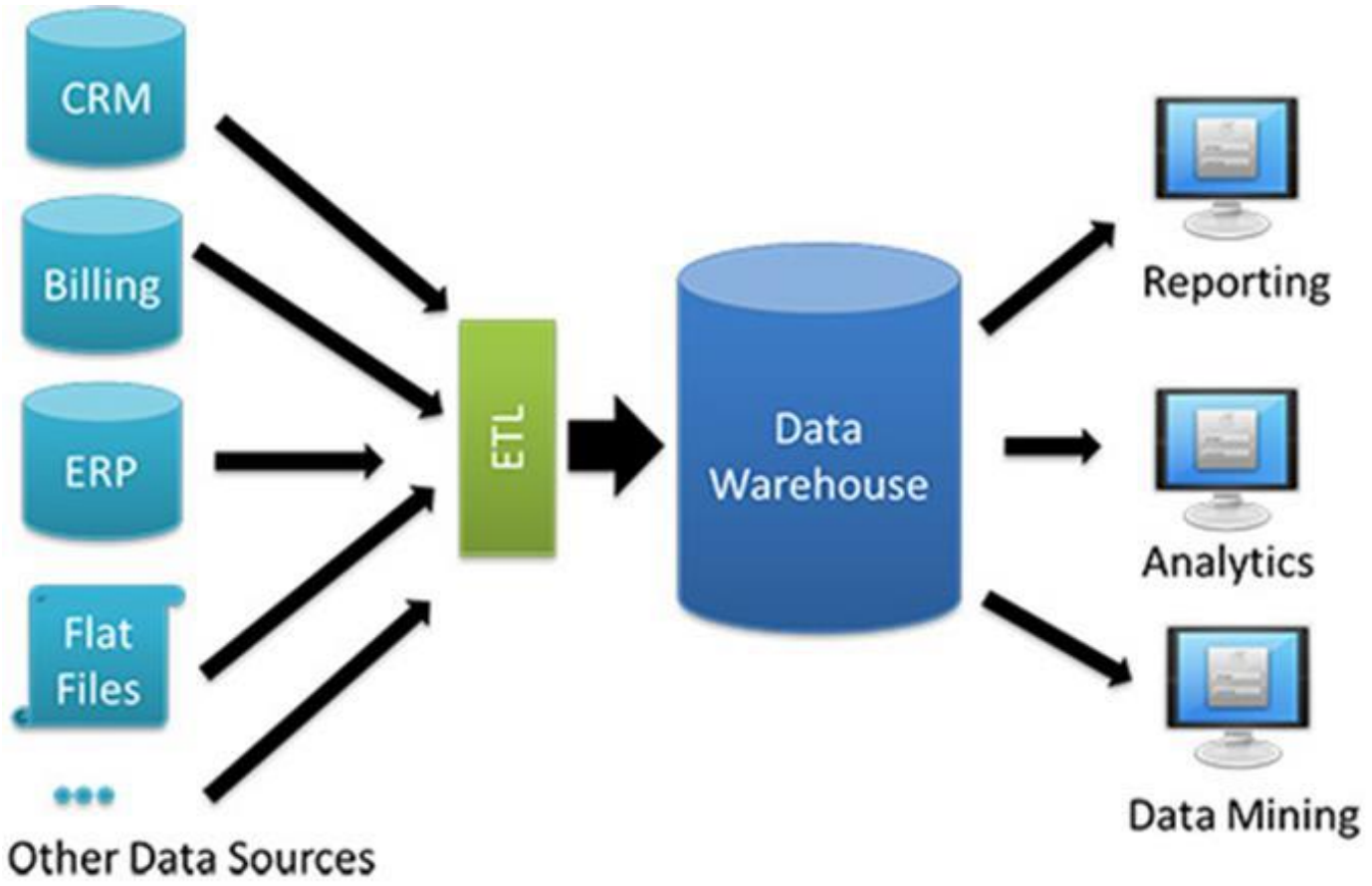
Data logic is implemented in the database procedure language

Prevalent in two-tiered architecture

Suited for data integration kind of work where the amount of data movement is usually large.


Indexes

Data Warehousing



OLTP Challenges

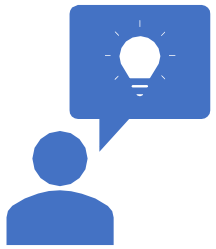
OLTP systems are not always good for handling aggregates over large amounts of data. OLTP should be doing quick short-live queries over transaction. Should not contend with OLAP requirements at the same time.

OLAP queries will tend to be quite complex when conducting analytics and reporting on data that is highly normalized.

Many times, we might want to integrate our business data with data from other sources. If we did this in OLTP, we will have a management issue of missing disparate data with our business data model.

Most OLTP are not designed to keep history. So performing analytics over historical data will face the shortcoming of having updated the historical data in an OLTP system.

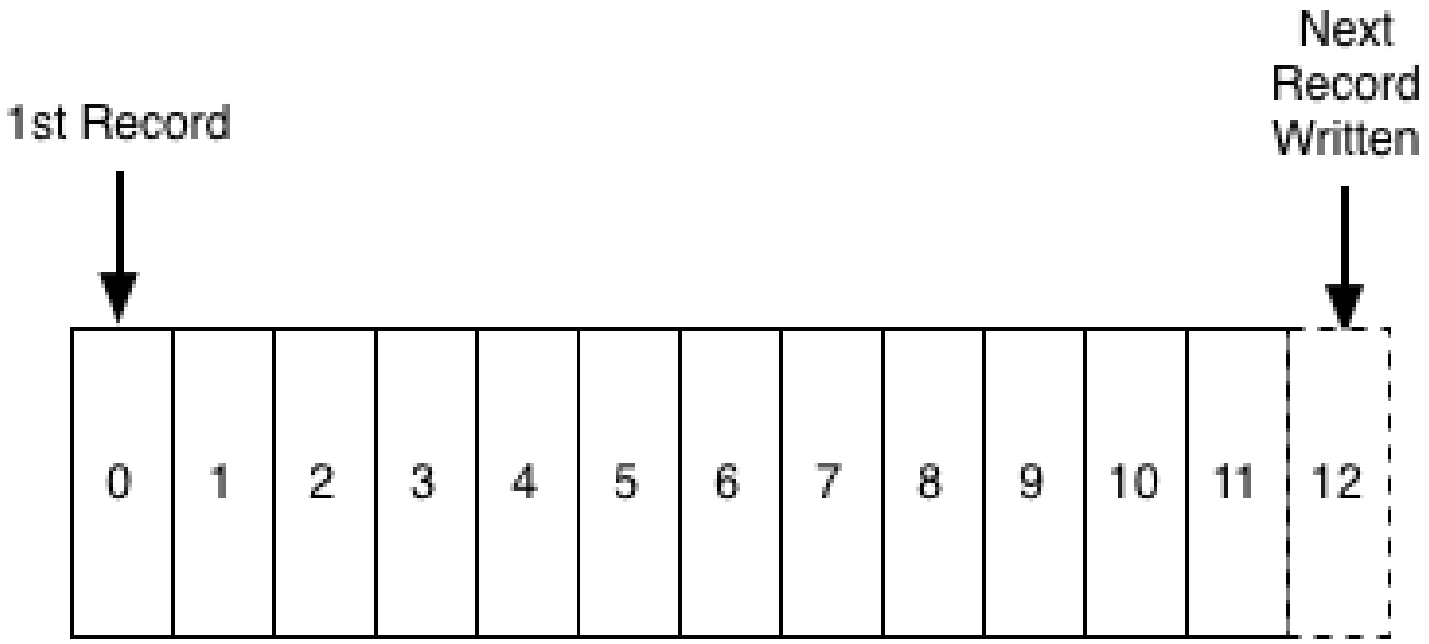
As data begins to grow, we need different environments that we can tune separately for the various kinds of workload.



Analytical Processing

- Aggregation Queries
- Window Queries

Logs



Relational Database Wrap-up

Gwen (Chen) Shapira
@gwenshap

Follow

I thought that every engineer knows SQL. Today I discovered that since we hire people who spent most of their career optimizing storage drivers and kernels, they actually don't know SQL. I hope it isn't too challenging to learn. Should be a friendly language, right?

8:20 PM - 18 Jun 2019

10 Retweets **149** Likes

60 10 149

Tweet your reply

Tanel Põder @TanelPoder · Jun 18
Replying to @gwenshap
Too friendly (makes me suspicious)

1 7

Tanel Põder @TanelPoder · Jun 18

writing two books on SQL can't say I miss it. Who wants to write 5 lines of SQL when you can write 30 lines of JavaScript to do the same thing?

5 7

Gwen (Chen) Shapira @gwenshap · 20h
You mean 3000 lines in C++?

2

dby @el_dby · Jun 18
Replying to @gwenshap
I would argue that In a world where analysts don't bother learning SQL (yes, it's a thing unfortunately) , engineers especially SREs will only pickup what's needed to solve a problem as itsbescakates to them.

I recall some tough mornings looking at redshift queries going WTAF

6

Timothy Perrett @timperrett · Jun 18
Replying to @gwenshap
Friendly language with absolutely no edge cases or strange grammar, implemented in exactly the same manner by all vendors. #Sarcasm

1 16

Ex falso quodlibet @philderome · 21h
write once run everywhere #doublesarcasm (files, tasks)

Nick Heudecker @nheudecker · Jun 18
Replying to @gwenshap
Many new devs don't want to learn SQL because they (incorrectly) see it as a