

# SQL FUNDAMENTALS

A collection of colorful geometric shapes including triangles, squares, circles, and a large circle outline, scattered across the dark blue background.

Steve Mitchell

# AGENDA

Databases and SQL

Anatomy of a Query

Order of Execution

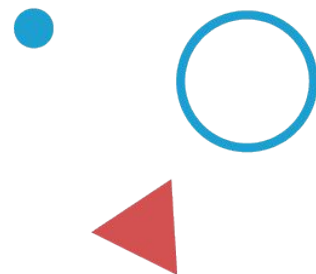
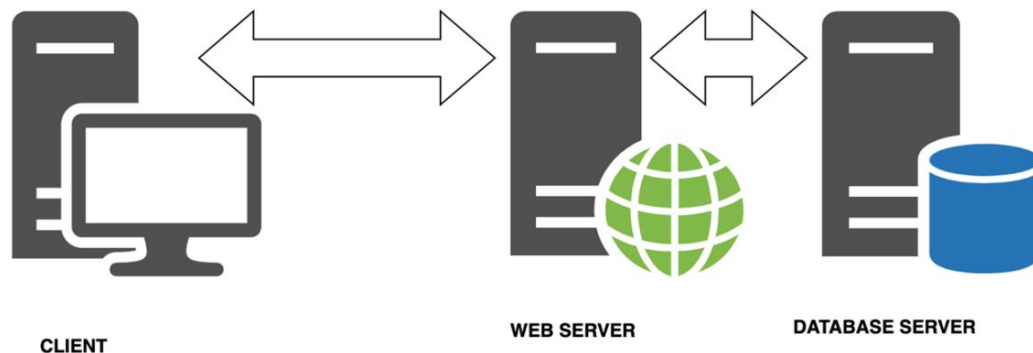
Debugging a Query

PostgreSQL Setup

# Databases and SQL

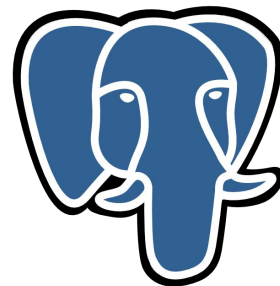
# What is a database?

- Organized collection of persistent data
- Info stored to be retrieved efficiently at a later time
- Third tier of web development architecture
- Different types of databases and architectures

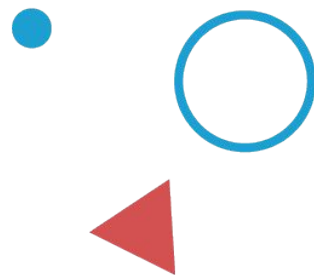


# Database Management System

- Software designed specifically for managing data in a database
- Manipulates data itself, data format, file structures
- Defines rules to validate and manipulate data using query language (SQL)
- We are using PGAdmin with PostgreSQL (open source)

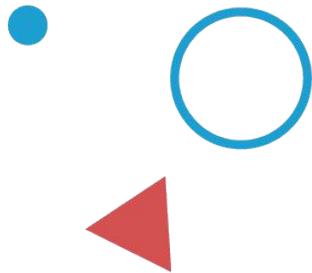


PostgreSQL

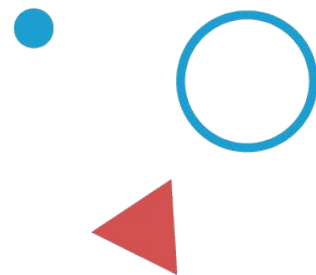
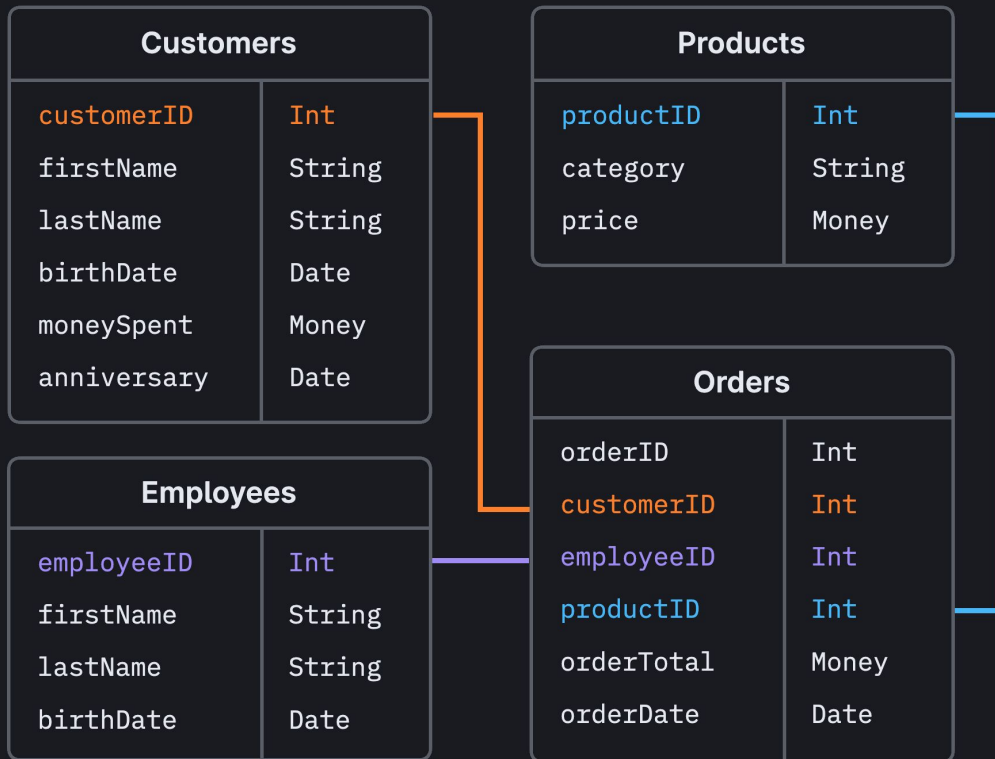


# Relational Database

- Multiple tables
- Tables must be related somehow
- Optimized to store and handle lots of large queries
- Cross platform - tabular data is usually stored and queried using SQL and it's a standard



# Why Relational?

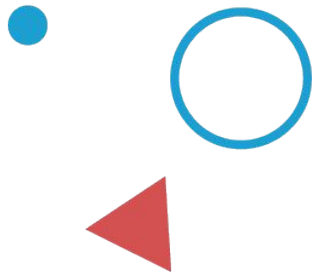


# PostgreSQL Data Types

<https://www.postgresql.org/docs/current/datatype.html>

Most common:

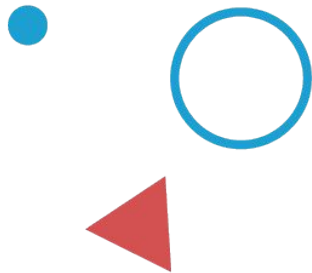
- **Integers (int):** whole numbers
- **Double precision (float):** number with decimals
- **Varchar:** varying character (text string)
- **Boolean:** true / false





# PostgreSQL Data Language

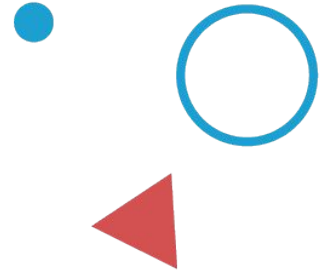
- **Definition (DDL)**
  - Create database
  - Delete database
  - Create, modify or delete table
- **Manipulation (DML)**
  - INSERT
  - UPDATE
  - DELETE
  - SELECT



# Anatomy of a Query

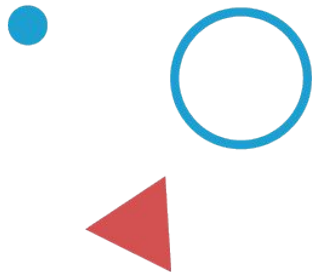
# Table Relation

- Tables have id columns, Primary Key (PK)
  - PK is unique identifier for each row
  - PK ensures that each row is in fact unique, part of data validation
- Tables have Foreign Keys (FK), containing id value of a table that it references
  - FK is further step in data validation
  - FK prevents manipulation errors
- ERD for database



# Anatomy of a Query

**SELECT** [ columns and functions ]  
[ **FROM** from\_item ]  
[ **WHERE** condition ]  
[ **GROUP BY** [ ALL | DISTINCT ] grouping\_element ]  
[ **HAVING** condition ]  
[ **ORDER BY** expression [ ASC | DESC ] ]  
[ **LIMIT** [ count | ALL ] ]  
[ **OFFSET** start [ ROW | ROWS ] ];

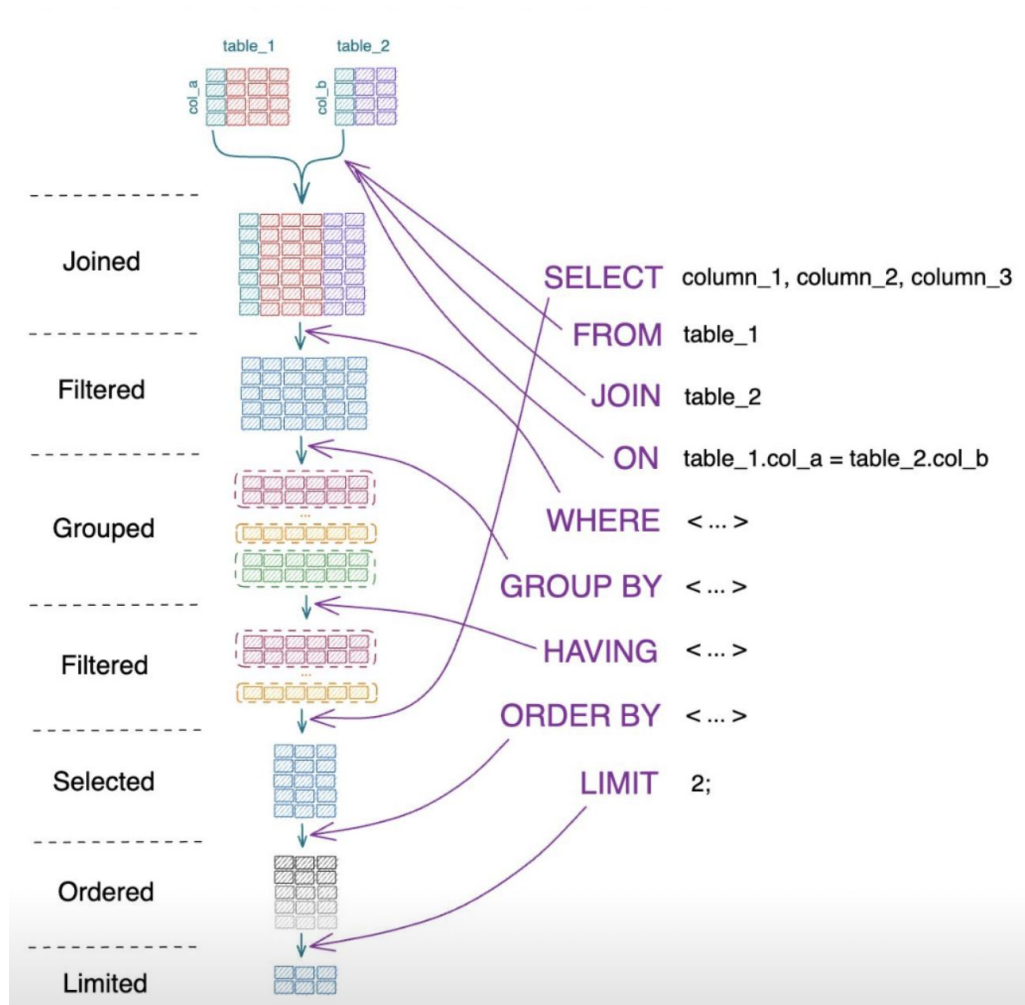


# Order of Execution

# Order of Execution

What the query looks like		How it's executed		Why it works this way
SELECT	▶	FROM	▶	SQL starts with which table your query is taking data from.
FROM	▶	WHERE	▶	This is how SQL filters on rows.
WHERE	▶	GROUP BY	▶	This is where your SQL query checks if you have an aggregation.
GROUP BY	▶	HAVING	▶	HAVING requires a GROUP BY statement.
HAVING	▶	SELECT	▶	Only after all these calculations have been made will SQL "SELECT" which columns you want to see returned.
ORDER BY	▶	ORDER BY	▶	This sorts the data returned.
LIMIT	▶	LIMIT	▶	Lastly, you can limit the number of rows returned.

# Order of Execution



# Let's Query Some Data!



# Debugging a Query

# Common Errors

- Error: syntax error at or near...
  - Locate the error
  - Check for typos
  - Check order of execution
  - Check brackets / semicolons
- Column does not exist
  - Check for typos
  - Check aliases
- Permission denied for relation...
  - Correct user / password / database?

