W5D1 - SQL Intro

LIGHTHOUSE LABS

AGENDA

- Why Databases
- DBMS
- Relational Databases
- SQL

LIGHTHOUSE LABS

Why databases?

slido

What problems are databases solving? Select all that applies

ⓘ Start presenting to display the poll results on this slide.

Third Tier of The Web Development Architecture

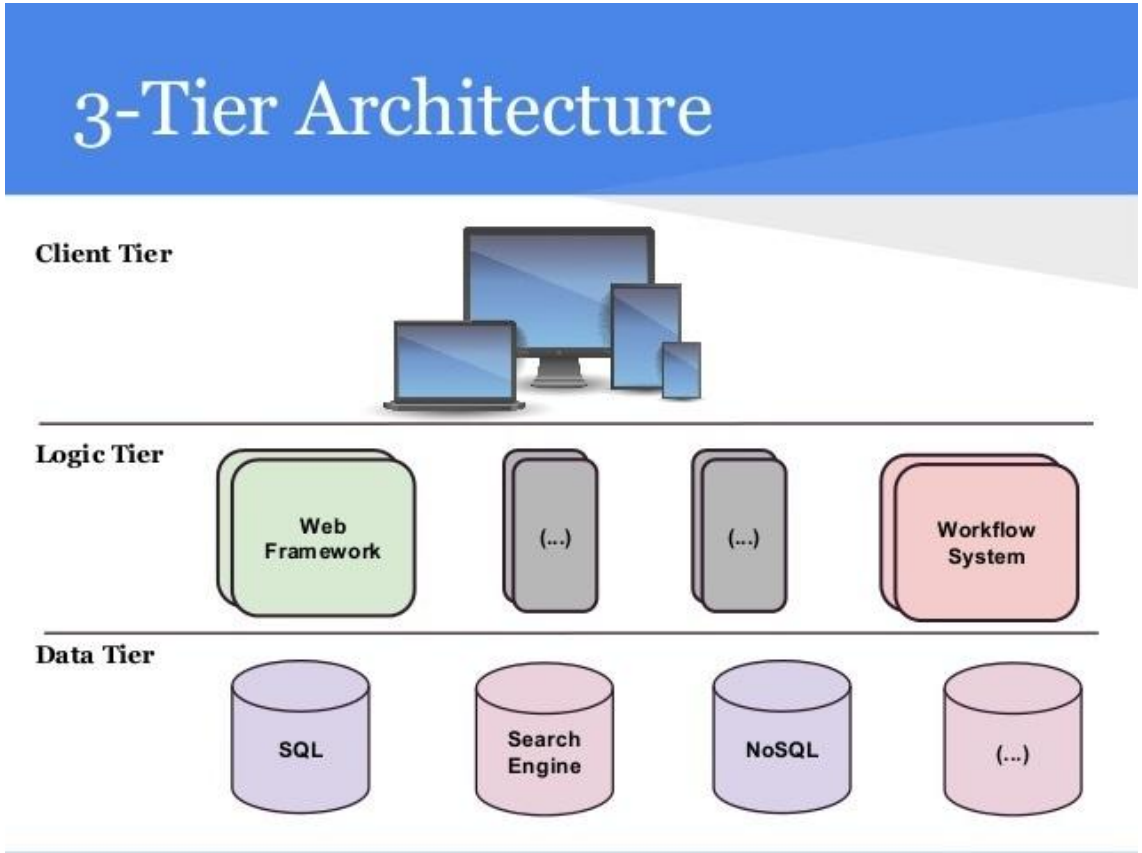


Image Reference

Database Management System (DBMS)

Software that provides an efficient storage mechanism that allows organization, manipulation, and protection of highly structured data.

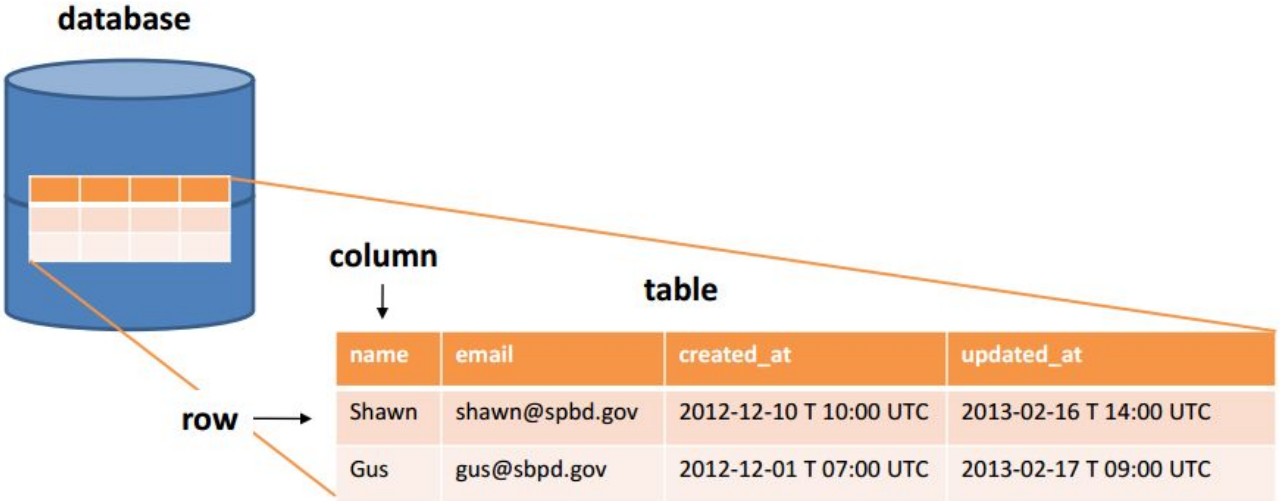
slido

What are the functions of a DBMS?

ⓘ Start presenting to display the poll results on this slide.

Relational Databases

- The most popular type of databases of the last few decades
- The data is broken down into tables with associations between them (relationships)
- Each table is like a spreadsheet with columns and rows
- We can query data with SQL



Source: Relational Databases for Dummies

SQL

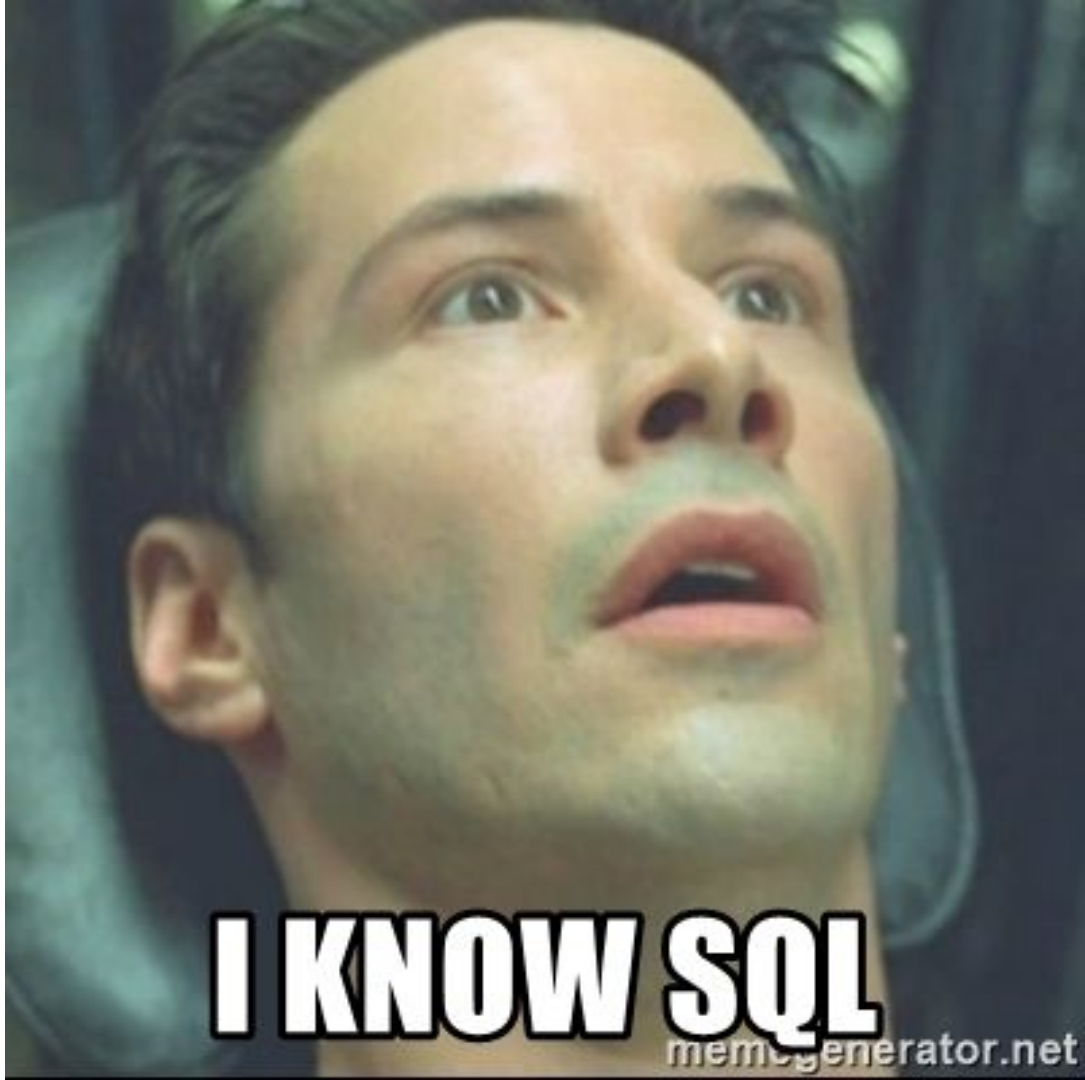


Image source

slido

What was SQL initially called?

ⓘ Start presenting to display the poll results on this slide.

SQL (Structured Query Language)

- Developed by IBM in the early 70s
- Initially called SEQUEL (Structured English Query Language)
- Declarative language (vs imperative)
  - We state only what we need
  - Abstract how to get it
- Grouped into DDL and DML

DDL - Data Definition Language

Create and modify the structure of a database

```
DROP TABLE IF EXISTS games;
DROP TABLE IF EXISTS platforms;

CREATE TABLE platforms(
  id SERIAL PRIMARY KEY,
  name TEXT NOT NULL,
  developer TEXT NOT NULL,
  release_date DATE);

CREATE TABLE games(
  id SERIAL PRIMARY KEY,
  title TEXT NOT NULL,
  description TEXT,
  release_date DATE,
  rating TEXT,
  genre TEXT,
  platform_id INTEGER REFERENCES platforms(id)
);
```

DML - Data Manipulation Language

Operations to manipulate the data:

- INSERT
- UPDATE
- DELETE
- SELECT

```
ALTER SEQUENCE platforms_id_seq RESTART WITH 1;
ALTER SEQUENCE games_id_seq RESTART WITH 1;


INSERT INTO platforms(name, developer, release_date) VALUES
('X Box One', 'Microsoft', '2013-11-22');
INSERT INTO platforms(name, developer, release_date) VALUES
('PlayStation 4', 'Sony', '2013-11-15');
INSERT INTO platforms(name, developer, release_date) VALUES
('Switch', 'Nintendo', '2017-03-03');
```

SQL Demo

LIGHTHOUSE LABS

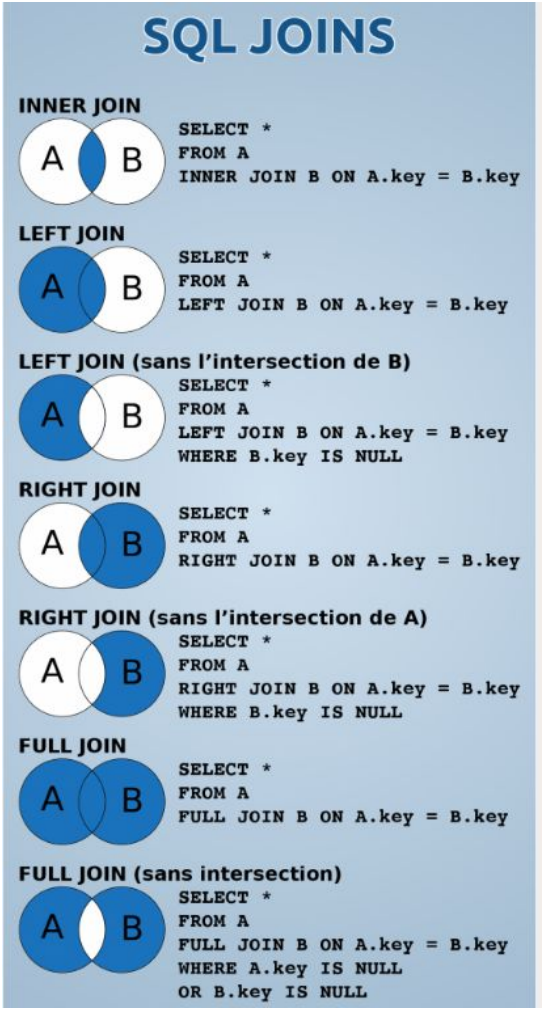
SELECT Statement

```
SELECT column list, function(), function(), ...
FROM table1
INNER JOIN table2
...
ON table1.col1 = table2.col2
...
WHERE criteria for row selection
[AND criteria for row selection]
[OR criteria for row selection]
GROUP BY column list
HAVING criteria for function results
ORDER BY column list;
```

JOINS

- We use joins when we need the data of more than one table

- There are different types of joins:
  - INNER JOIN
  - [LEFT || RIGHT] OUTER JOIN



Games DB

From the command-line:

```
> createdb games
> psql games < db/create.sql
> psql games < db/seeds.sql
```

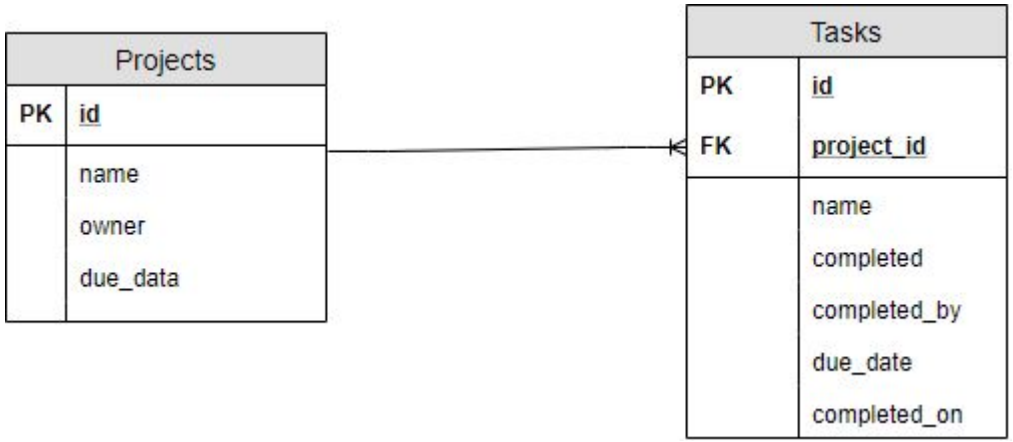
SELECT Queries:

- Basic SELECT statements
- DISTINCT
- WHERE
- ORDER BY
- LIMIT
- OFFSET
- IN
- Like
- Aggregate functions - `COUNT`, `SUM`, `AVG`, `MIN`,`MAX`
- Having
- Joins

SQL Exercise

LIGHTHOUSE LABS

SQL Exercise



[SQL Exercise gist]

Questions?

LIGHTHOUSE LABS