SQL Introduction

Module 5 Week 11

Notes Repo: https://github.com/C-Shi/lhl-flex-lecture

Learning Objectives

Database Concept and Database Software

Relational Database Concept

Structured Query Language or SQL

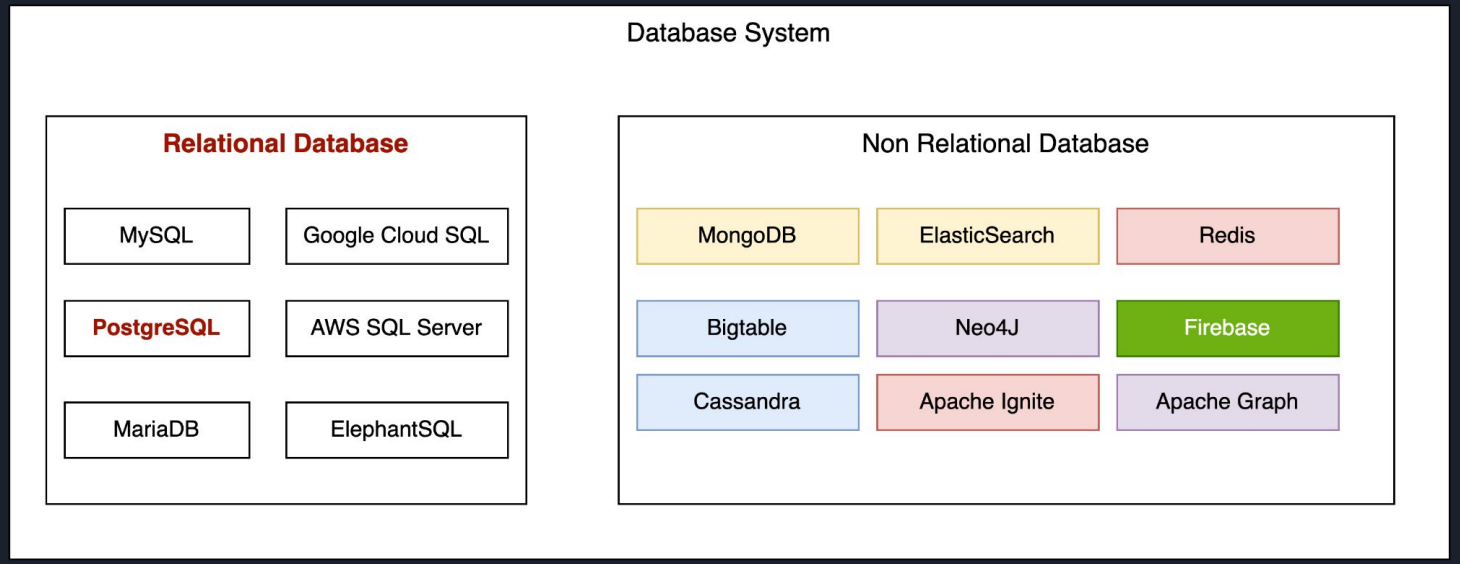
- Table management query
- Data management query

What is Database

An organized collection of information stored electronically

A program that interact and manipulate data is called Database Management System(DBMS)

Data and Database Management System together are often referred to as Database



Relational Database

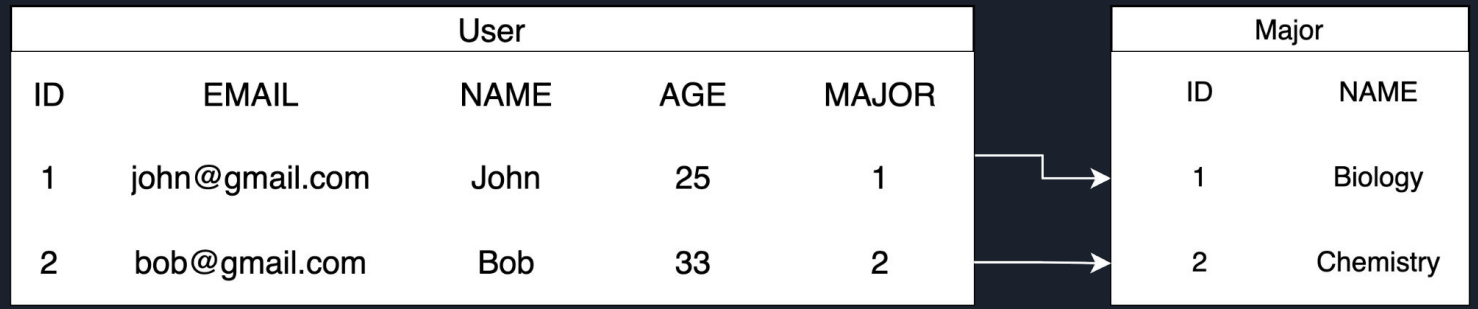
A collection of data pre-defined with relationships

Utilize SQL to retrieve and maintain data

The fundamental of relational database is **TABLE**

Table contains sets of **COLUMN**

Every piece of data is stored as a **ROW**



User

ID	EMAIL	NAME	AGE	MAJOR
1	john@gmail.com	John	25	1
2	bob@gmail.com	Bob	33	2

Major

ID	NAME
1	Biology
2	Chemistry

Exercise 1

There are 3 Students in a college

Each Students has a student ID, name, email

The college offer a few majors, and each student has to choose a major

A few year later, the college decide to rename a major

Before Actual SQL Language

students			
id	name	email	major_id
1	John	john@gmail.com	1
2	Bob	bob@gmail.com	2
3	David	david@gmail.con	1

majors	
id	name
1	Computer Science
2	Economy

SQL Terms	
Table	A fundamental Collection of data
Column	A field in a table that specify a piece of information
Row	One complete record
Primary key	A field (fields) that uniquely identify a record
Foreign key	A field (fields) that refer to the primary key in another table
*One-to-Many	A row in table A has many matching in table B. But a row in table B has only one matching in table A

psql

A terminal-based front end interactive interface to connect to PostgreSQL

Command	**Usage**
psql	To enter psql interface
\l	List all databases
\c database_name	To use a certain database
\dt	List all tables
\d table_name	Describe a table
\q	Quit psql

SQL statements for Tables

CREATE	CREATE TABLE IF NOT EXISTS table_name (
	  column_name_1 datatype column_constraint,
	  column_name_2 datatype column_constraint
	)

MODIFY	ALTER TABLE table_name action column_name [datatype column_constraint]
DELETE	DROP TABLE IF EXISTS table_name

Exercise 2

1. Create a Table called **majors**, containing an auto incremented **id** and a **name**;
2. Create a Table called **students**, containing an auto incremented **id**, **name** and **email**.
3. Modify **students** table, add a column called **major_id**, that is a referencing the id of majors table
4. Modify **students** table, add a column called **year**
5. Drop **students** table
6. Drop **majors** table

SQL Statements for Data

CREATE	INSERT INTO table_name (column_1, column_2, …) VALUES (value_1, value_2, …), (value_3, value_4, …)
READ	SELECT column_names FROM table_name WHERE conditions
UPDATE	UPDATE table_name SET column_name = value WHERE conditions
DELETE	DELETE FROM table_name WHERE conditions

Exercise 2

1. Add a major called **Psychology** to **majors** table
2. Add a **second year** student called **Travis**, whose email is **travis@gmail.com**. His gpa is **3.2** and he does not have a major yet.
3. Pull report for all **students** **name** and **email**.
4. Pull report for all **third and fourth year students**
5. Pull report for the 3 students with **highest GPA**
6. Modify John's email to **john@hotmail.com**
7. Remove student **Ryan** from the table
8. Pull report for all students' name and major

Data from multiple table - JOIN

- To select data from multiple tables
- One table's primary key join with another table's foreign key

```
SELECT * FROM table_A JOIN table_B ON table_A.id == table_B.a_id;
```

- JOIN TYPE

SQL Aggregate Function

Functions to compute a single result from multiple rows

Some common aggregate functions are:

AVG	Calculate the average
COUNT	To count the total number of results
SUM	To summarize a column
MIN	Get the minimum value of a column
MAX	Get the maximum value of a column

Exercise 3

- Calculate the average GPA for all students
- Get the number of students in their last (4) year
- Get the student with highest GPA from second year
- Get the average GPA for every major

Grouping Result

Grouping the rows that has same value in one or more columns into one summary column

GROUP BY often used with Aggregate functions

Use HAVING to filter the results

```
SELECT major_id, AVG(gpa) FROM students
GROUP BY major_id HAVING AVG(gpa) > 3.0
```

GET the average GPA for Economy Major

GET all majors that have average GPA > 3.0

WHERE or HAVING

WHERE and HAVING may be used interchangeably in some situation

	WHERE	HAVING
usage	To filter data	To filter data
position	Before GROUP BY	After GROUP BY
act on	Database column	Grouped result
In result	Not necessary	YES

```
SELECT majors.name, AVG(gpa) FROM students JOIN majors ON students.major_id = majors.id
GROUP BY majors.name HAVING majors.name = 'Economy';

SELECT majors.name, AVG(gpa) FROM students JOIN majors ON students.major_id = majors.id
WHERE majors.name = 'Economy'
GROUP BY majors.name;
```