# Data Structures/Big Data

Chulalongkorn University
School of Integrated Innovation
Fall 2024
Marko Niinimaki, marko.n@chula.ac.th

Today's main topic: Intro to databases

# Last week: Reading files, CSV

```
Fname,Minit,Lname,Ssn,Bdate,Address,Gender,Salary,Super_ssn,Dept
John,B,Smith,123456789,1965-01-09,731-Fondren-Houston-TX,M,30000,333445555,5
Franklin,T,Wong,333445555,1955-12-08,638-Voss-Houston-TX,M,40000,888665555,5
Alicia,J,Zelaya,999887777,1968-01-19,3321-Castle-Spring-TX,F,25000,987654321,4
Jennifer,S,Wallace,987654321,1941-06-20,291-Berry-Bellaire-TX,F,43000,888665555,4
Ramesh,K,Narayan,666884444,1962-09-15,975-Fire-Oak-Humble-TX,M,38000,333445555,5
Joyce,A,English,453453453,1972-07-31,5631-Rice-Houston-TX,F,25000,333445555,5
Ahmad,V,Jabbar,987987987,1969-03-29,980-Dallas-Houston-TX,M,25000,987654321,4
James,E,Borg,888665555,1937-11-10,450-Stone-Houston-TX,M,55000,NULL,1
```

```
import csv

with open('company.csv') as csv_file:
   csv_reader = csv.reader(csv_file, delimiter=',')
   line_count = 0
   for row in csv_reader:
       line_count += 1
       if line_count > 1:
           print(row[2]) #lname in pos 2 (start 0)
```

## Last week: Reading files, JSON

```
{
  "name": "John Doe",
  "age": 30,
  "email": "johndoe@example.com",
  "hobbies": ["reading", "traveling", "coding"],
  "address": {
     "street": "123 Maple Street",
     "city": "Anytown",
     "state": "CA",
     "postal_code": "90210"
  }
}
```

```
import json

with open("person1.json", "r") as read_file:
   data = json.load(read_file)
   print(data["name"])
```

## Last week: Reading files, JSON

```
[
  {
     "name": "John Doe",
     "age": 30,
     "email": "johndoe@example.com",
     "skills": ["Python", "Data Analysis",
"Machine Learning"]
  },
  {
     "name": "Jane Smith",
     "age": 28,
     "email": "janesmith@example.com",
     "skills": ["JavaScript", "Web
Development", "React"]
  },
  {
     "name": "Michael Brown",
     "age": 35,
     "email":
"michaelbrown@example.com",
     "skills": ["Java", "Spring Boot",
"Microservices"]
```

```
import json

with open("persons.json", "r") as read_file:
   persons = json.load(read_file)
   for person in persons:
       if 'Python' in person['skills']:
           print(person['name'])
```

## Where is data? In databases

Definition of a database (Oxford English Dictionary):

A large collection of information that has been coded and stored in a computer in such a way that it can be extracted under a number of different category headings.

And a database management system DBMS (Prof Widom, Stanford):

A DBMS provides efficient, reliable, convenient and safe multi-user storage of and access to massive amounts of persistent data.

## Where is data? In databases

Who built databases and why?

Example 1: https://en.wikipedia.org/wiki/Sabre_(travel_reservation_system) IBM for American Airways, 1960.

Yes, before SABRE, flight reservations were really done by phone, pen & paper -> slow, prone to errors.

SABRE type of data: format very strict and regular. Strings and numbers.

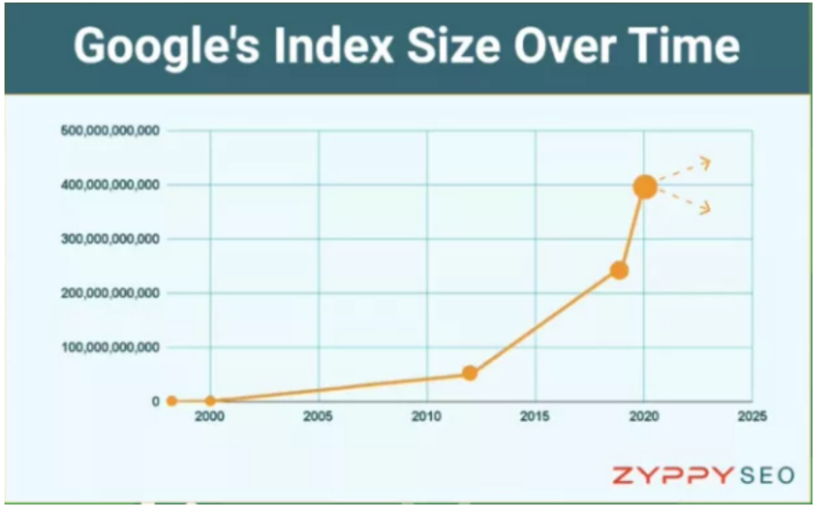
# Where is data? In databases

Who built databases and why?

Example 2: The Google Search Index

Contains information about ca. 400 billion documents. The index builder (Google web crawler) checks/indexes about 4 billion sites daily.

Type of data: textual, but includes metrics like "page rank" (the page's or site's reputation).



An index can be described as a dictionary of word/phrase → list of pages.

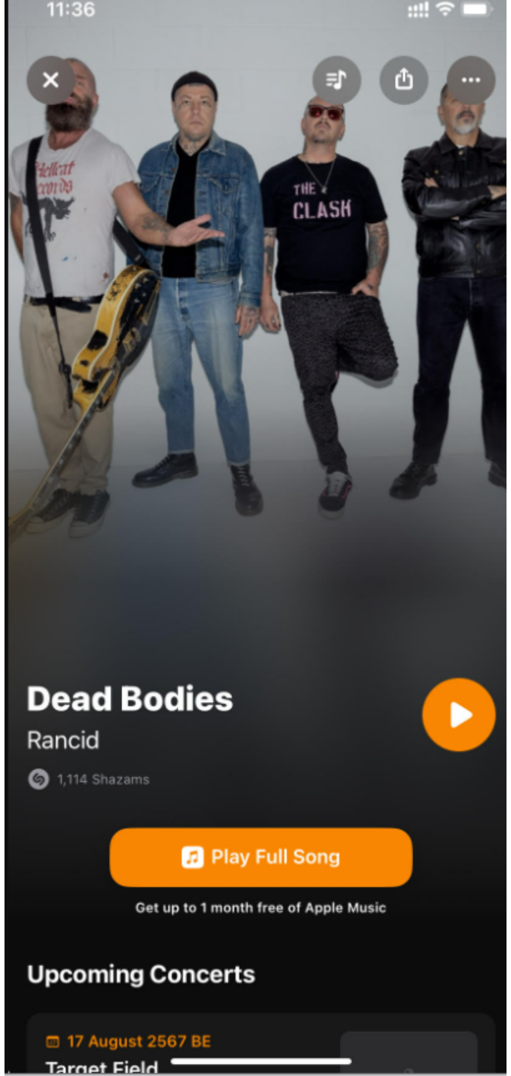
## Where is data? In databases

Who built databases and why?

Example 3: Shazam audio

Estimated containing 11 million songs (or their "fingerprints").

Type of data: audio, images, concert information.



Dead Bodies

Rancid

1,114 Shazams

Get up to 1 month free of Apple Music

Upcoming Concerts

17 August 2567 BE

Target Field

# Where is data? In databases

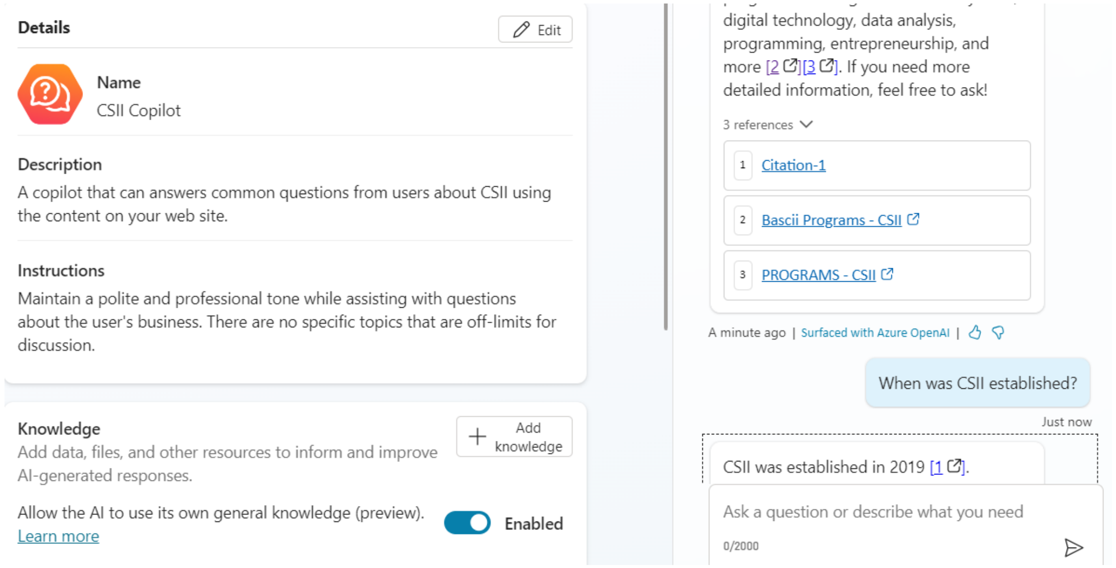
Who built databases and why?

Example 4: RAG

(Retrieval Assisted Generation): Add local knowledge about a specific subject to a large language model.

Type of data: Text, spreadsheets,..

This is a "vector database".



Details

Edit

Name

CSII Copilot

Description

A copilot that can answers common questions from users about CSII using the content on your web site.

Instructions

Maintain a polite and professional tone while assisting with questions about the user's business. There are no specific topics that are off-limits for discussion.

Knowledge

Add knowledge

Add data, files, and other resources to inform and improve AI-generated responses.

Allow the AI to use its own general knowledge (preview). Learn more

Enabled

digital technology, data analysis, programming, entrepreneurship, and more [2][3]. If you need more detailed information, feel free to ask!

3 references

1 Citation-1

2 Bascii Programs - CSII

3 PROGRAMS - CSII

A minute ago | Surfaced with Azure OpenAI |

When was CSII established?

Just now

CSII was established in 2019 [1].

Ask a question or describe what you need

0/2000

# Where is data? In databases

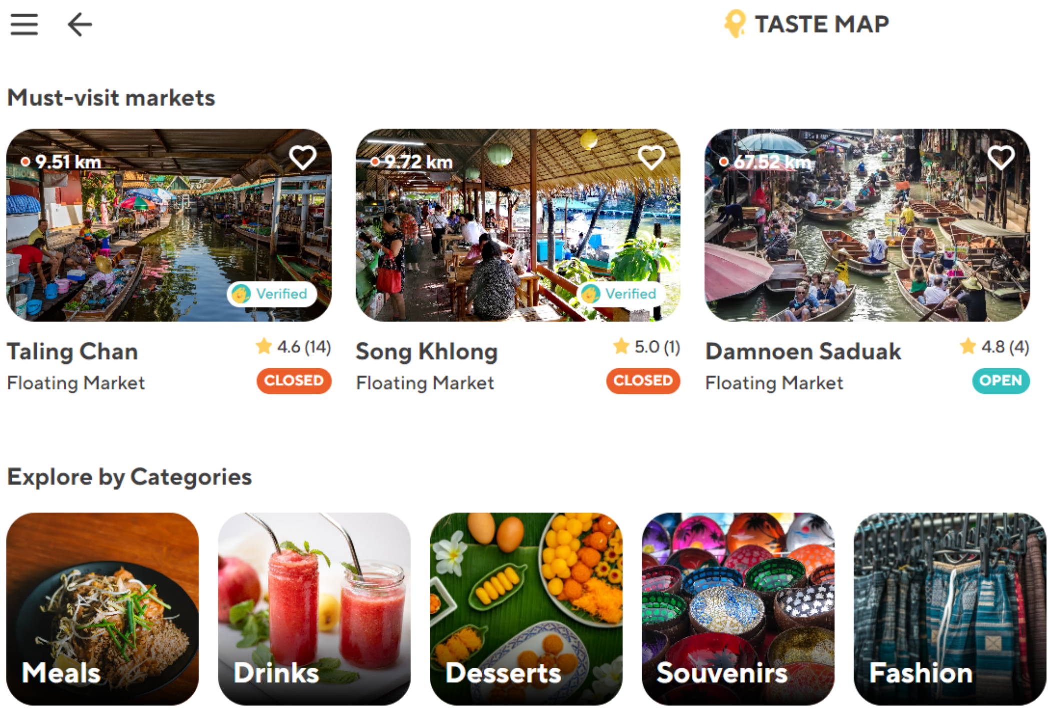
Who built databases and why?

Example 5: You

Most web and mobile applications get their data from and store their data into a database.

Type of data: almost anything.

Often a "noSQL" database.



## Where is data? In databases

The video https://www.youtube.com/watch?v=W2Z7fbCLSTw mentions "7 database paradigms"
Key-value
Wide column
Document
Relational
Graph
Text search engine
Multi-model
+ some types may still be missing: multimedia DB's.

## Why do we use databases/DBMS's? Why not just files?

1 Speed: Remember our CSV file reader? A Python program that reads a file and prints the line where the name matches user's input. Test with a 1000 lines file.

time python3 read_company_input_line.py Linus
0.192 s

The same operation with a database + database management system:
0.01 s

2 Control: Standard operations for creating/reading/updating/deleting data items (CRUD), other "nice" properties for data management like atomicity, consistency, isolation, and durability (ACID – but this is not provided by all DBMS's).

## Today’s main topics

Where is data? In programs/files/**databases**/...

What are relational databases?

## Relational databases and the relational model

| Fname | Sname | ssn | date_of_birth |
|---|---|---|---|
| 'John' | 'Smith' | **123456789** | '1965-01-09' |
| 'Franklin' | 'Wong' | **333445555** | '1965-12-08' |
| 'Alicia' | 'Zelaya' | **999887777** | '1968-01-19' |
| 'Jennifer' | 'Wallace' | **987654321** | '1941-06-20' |
| 'Ramesh' | 'Narayan' | **666884444** | '1962-09-15' |
| 'Joyce' | 'English' | **453453453** | '1972-07-31' |
| 'Ahmad' | 'Jabbar' | **987987987** | '1969-03-29' |
| 'James' | 'Borg' | **888665555** | '1937-11-10' |

Project: Get data from a certain column (or multiple columns) by names.

Like ssn here.

# Relational databases and the relational model

| Fname | Sname | ssn | date_of_birth |
|---|---|---|---|
| **'John'** | **'Smith'** | **123456789** | **'1965-01-09'** |
| **'Franklin'** | **'Wong'** | **333445555** | **'1965-12-08'** |
| 'Alicia' | 'Zelaya' | 999887777 | '1968-01-19' |
| 'Jennifer' | 'Wallace' | 987654321 | '1941-06-20' |
| 'Ramesh' | 'Narayan' | 666884444 | '1962-09-15' |
| 'Joyce' | 'English' | 453453453 | '1972-07-31' |
| 'Ahmad' | 'Jabbar' | 987987987 | '1969-03-29' |
| 'James' | 'Borg' | 888665555 | '1937-11-10' |

Select: Get some rows by criteria

Example: rows such that year of birth is 1965

# Relational databases and the relational model

Employees

| Fname | Sname | dno |
|---|---|---|
| 'Franklin' | 'Wong' | 5 |
| 'Alicia' | 'Zelaya' | 4 |
| 'James' | 'Borg' | 1 |

Departments

| Dname | Dnum |
|---|---|
| 'Research' | 5 |
| 'Administration' | 4 |
| 'Headquarters' | 1 |

Employees x Departments

| Fname | Sname | dno | Dname | Dnum |
|---|---|---|---|---|
| 'Franklin' | 'Wong' | 5 | 'Research' | 5 |
| 'Franklin' | 'Wong' | 5 | 'Administration' | 4 |
| 'Franklin' | 'Wong' | 5 | 'Headquarters' | 1 |
| 'Alicia' | 'Zelaya' | 4 | 'Research' | 5 |
| 'Alicia' | 'Zelaya' | 4 | 'Administration' | 4 |
| 'Alicia' | 'Zelaya' | 4 | 'Headquarters' | 1 |
| 'James' | 'Borg' | 1 | 'Research' | 5 |
| 'James' | 'Borg' | 1 | 'Administration' | 4 |
| 'James' | 'Borg' | 1 | 'Headquarters' | 1 |

Cartesian product: Combine everything

## What's the point?

A request like "give me the employees who work for Research" would be:

Rel1 = Employees x Departments

Rel2 = select (Dno = Dnum, dname = 'Research') Rel1

```
Rel2
Fname       Sname dno       Dname       Dnum
'Franklin'    'Wong'    5   'Research'        5
```

A request like "give me the surnames of the employees who work for research" would be:

Rel1 = Employees x Departments

Rel2 = select (Dno = Dnum, dname = 'Research') Rel1

Rel3 = project Sname Rel2

## How did this ..

https://en.wikipedia.org/wiki/Edgar_F._Codd

Dr Codd invented relational databases, did not invent SQL

The keyword "select" in SQL is not the same as the select operation in the previous slides.

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

But both Codd and the inventors of SQL worked for IBM

IBM Db2 and Oracle (https://en.wikipedia.org/wiki/Oracle_Database) were among the first SQL databases.

## Next

Relational database exercises but let's build our database first using PythonAnywhere.

If we have time: The SQL language cheat sheet: https://learnsql.com/blog/sql-basics-cheat-sheet/sql-basics-cheat-sheet-a4.pdf

Next week's topic: SQL (visiting lecturer).