

# 1

## Introduction to Databases

CSF2600700 - BASIS DATA





## Acknowledgements

This slide is a modification to supplementary slide of  
“Database System”, 6th edition, Elmasri/Navathe, 2011: **Chapter 1 Introduction to Databases**  
and “**Introduction to Databases**” used in “Basis Data” course in academic years 2018/2019 in the Faculty  
of Computer Science, Universitas Indonesia

100

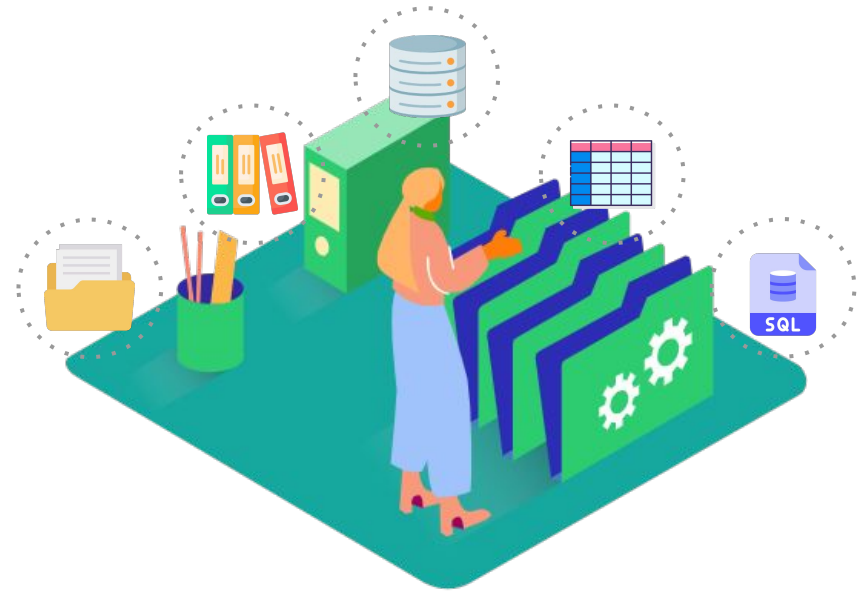
## 1. Introduction

## 2. An Example

### 3. Actors on the scene

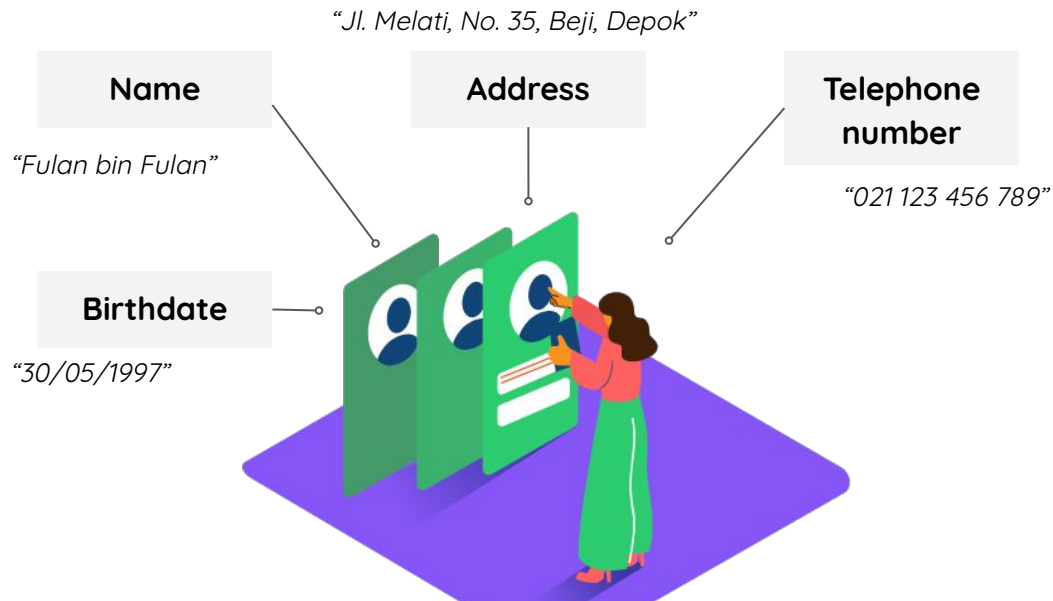
#### 4. Workers behind the scene

## 5. When to use database



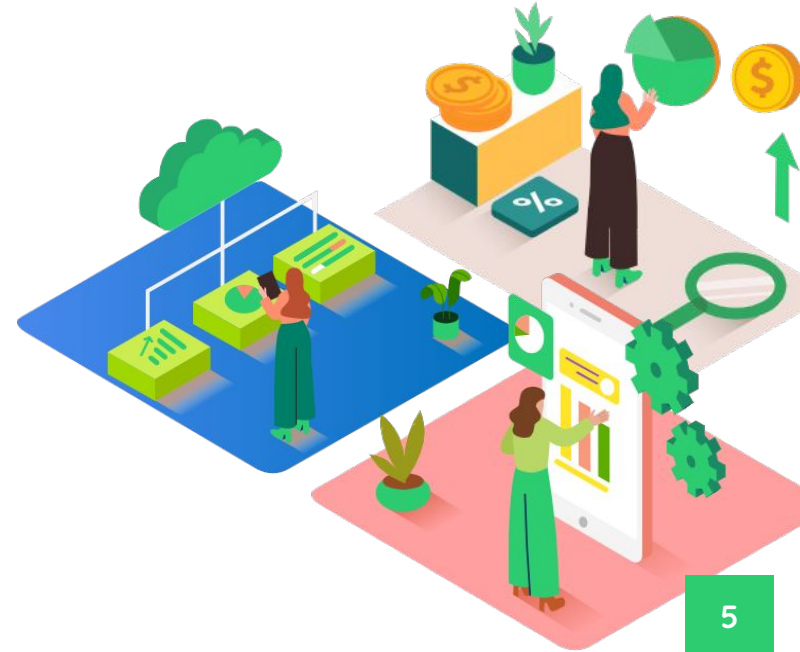
# Definition

- **Collection** of related data
- Known facts that can be recorded and that have implicit meaning



# Database Properties

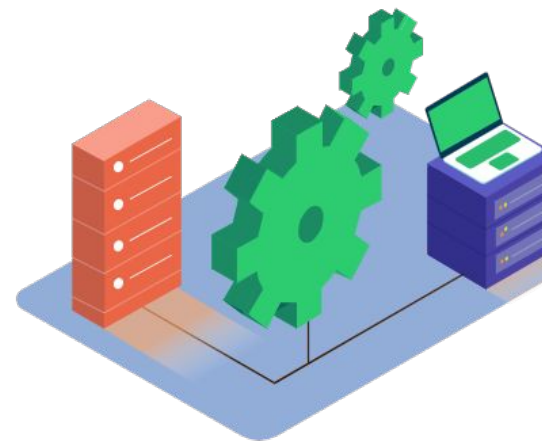
- **Miniworld** or **universe of discourse (UoD)**:  
Represents some aspect of the real world
- Logically coherent collection of data with inherent meaning
- Built for a specific purpose



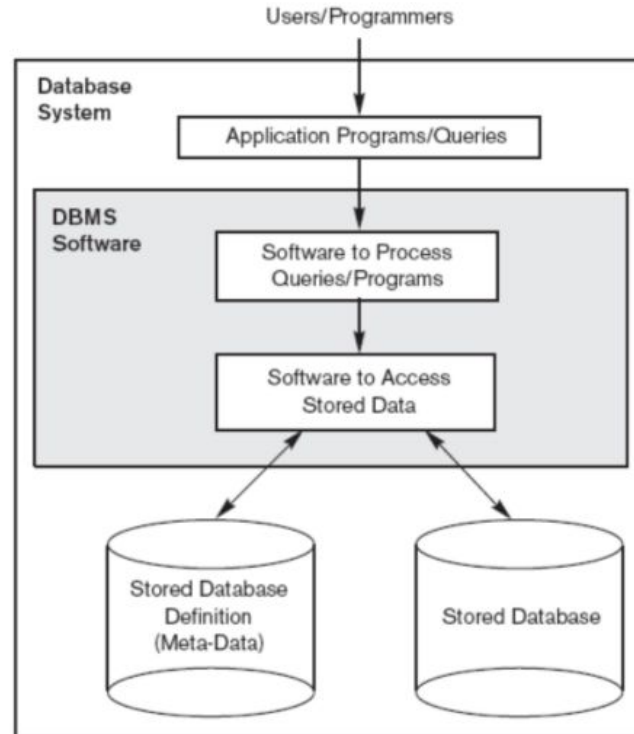
# Database Management System (DBMS)

- Collection of programs
- Enables users to **create** and **maintain** a database

## Example



# Database System

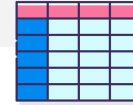


**Figure 1.1**  
A simplified database  
system environment.

# Examples

## Traditional database applications

Store **textual** or **numeric** information



## Multimedia databases

Store **images**, **audio** clips, and **video** streams digitally



## Geographic information systems (GIS)

Store and analyze **maps**, **weather data**, and **satellite images**

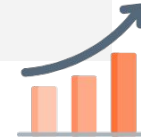




# Defining a Database

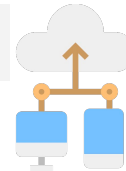
## Data warehouses and online analytical processing (OLAP) systems

Extract and analyze useful business information from **very large databases**  
Support **decision making**.



## Real-time and active database technology

Control industrial and manufacturing processes



# Defining a Database

Specify the data types, structures, and constraints of the data to be stored

## Meta-data

- Database **definition** or **descriptive information**
- Stored by the DBMS in the form of a database catalog or dictionary

### RELATIONS

Relation_name	No_of_columns
STUDENT	4
COURSE	4
SECTION	5
GRADE_REPORT	3
PREREQUISITE	2

### COLUMNS

Column_name	Data_type	Belongs_to_relation
Name	Character (30)	STUDENT
Student_number	Character (4)	STUDENT
Class	Integer (1)	STUDENT
Major	Major_type	STUDENT
Course_name	Character (10)	COURSE
Course_number	XXXXNNNN	COURSE
....	....	....
....	....	....
....	....	....
Prerequisite_number	XXXXNNNN	PREREQUISITE

*Note:* Major\_type is defined as an enumerated type with all known majors.  
XXXXNNNN is used to define a type with four alpha characters followed by four digits.

**Figure 1.3**  
An example of a database catalog for the database in Figure 1.2.

## Other Terms

### **Manipulating** a database

- Query and update the database miniworld
- Generate reports

### **Sharing** a database

- Allow multiple users and programs to access the database simultaneously

### **Application Program**

- Accesses database by sending queries to DBMS

### **Query**

- Causes some data to be retrieved

### **Transaction**

- May cause some data to be read and some data to be written into the database

### **Protection** includes

- System protection
- Security protection

### **Maintain** the database system

- Allow the system to evolve as requirements change over time

# Outline

Introduction

**An Example**

Actors on the scene

Workers behind the scene

When to use database



# An Example

## UNIVERSITY Database

Information concerning students, courses, and grades in a university environment



### Data records

STUDENTS

GRADE\_REPORT

COURSE

PREREQUISITE

SECTION



## An Example (Continued)

### Data records

STUDENTS	GRADE_REPORT
COURSE	PREREQUISITE
SECTION	



**Specify structure** of records of each file by specifying data type for each data element:

- String of alphabetic characters
- Integer
- Etc.



## An Example (Continued)



### Construct UNIVERSITY database

- **Store data** to represent each student, course, section, grade report, and prerequisite as a record in appropriate file
- **Relationships** among the records
- **Manipulation** involves querying and updating



# An Example (Continued)

**STUDENT**

Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS

**COURSE**

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS

**SECTION**

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	07	King
92	CS1310	Fall	07	Anderson
102	CS3320	Spring	08	Knuth
112	MATH2410	Fall	08	Chang
119	CS1310	Fall	08	Anderson
135	CS3380	Fall	08	Stone

**GRADE\_REPORT**

Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

**PREREQUISITE**

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

**Figure 1.2**  
A database that stores  
student and course  
information.





## An Example (Continued)



### Examples of Queries

- Retrieve the transcript
- List the names of students who took the section of the 'Database' course offered in fall 2008 and their grades in that section
- List the prerequisites of the 'Database' course



### Examples of Updates

- Change the class of 'Smith' to sophomore
- Create a new section for the 'Database' course for this semester
- Enter a grade of 'A' for 'Smith' in the 'Database' section of last semester



# Outline

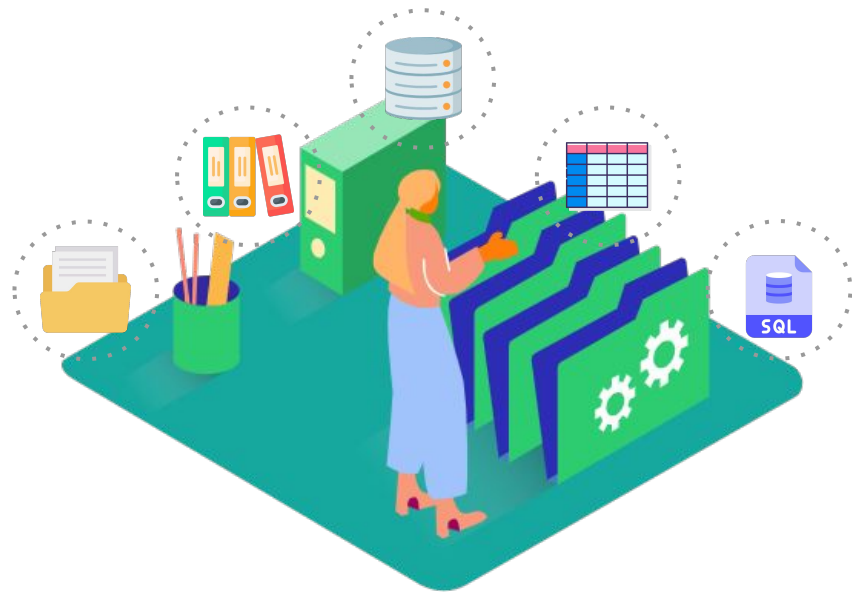
Introduction

An Example

Actors on the scene

Workers behind the scene

When to use database



# Actors on the Scene



## Database administrators (DBA)

are responsible for:

- Authorizing **access** to the database
- Coordinating and **monitoring** its use
- Acquiring software and hardware **resources**



## Database designers

are responsible for:

- **Identifying the data** to be stored
- Choosing appropriate **structures** to represent and store this data



## End users

People whose jobs require access to the database.

- **Naive or parametric** end users
- **Casual** end users
- **Sophisticated** end users
- **Standalone** users

## Actors on the Scene



### System analysts

determine **requirements** of end users



### Application programmers

**implement** these specifications as programs

# Outline

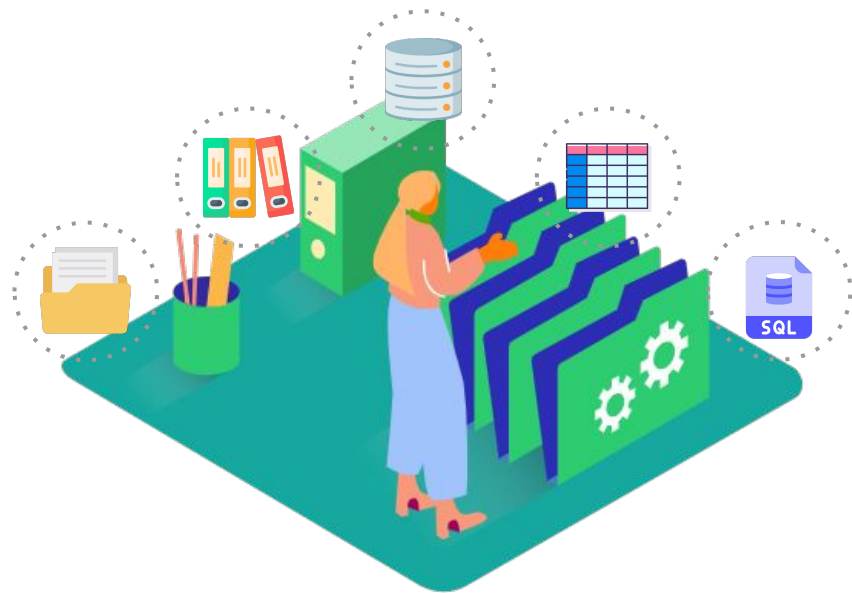
Introduction

An Example

Actors on the scene

Workers behind the scene

When to use database

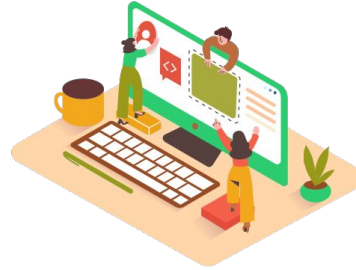


## Workers Behind the Scene



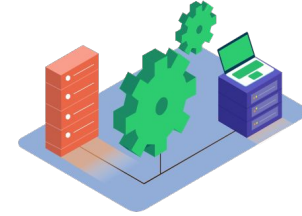
### DBMS system designers and implementers

Design and implement the **DBMS** modules and interfaces as a software package



### Tools developers

Design and implement **tools**



### Operators and maintenance personnels

Responsible for running and maintenance of hardware and **software environment** for database system

# Outline

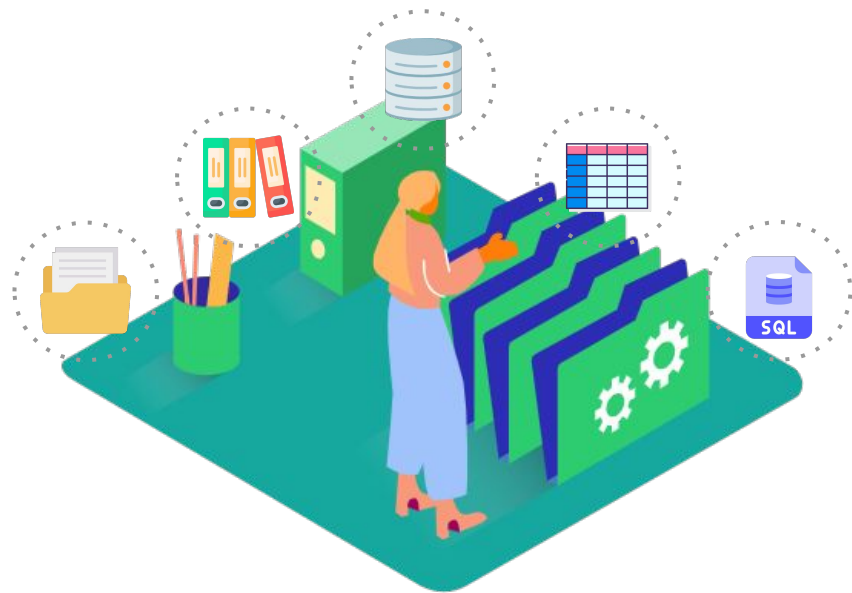
Introduction

An Example

Actors on the scene

Workers behind the scene

When to use database



## When we need to use database?

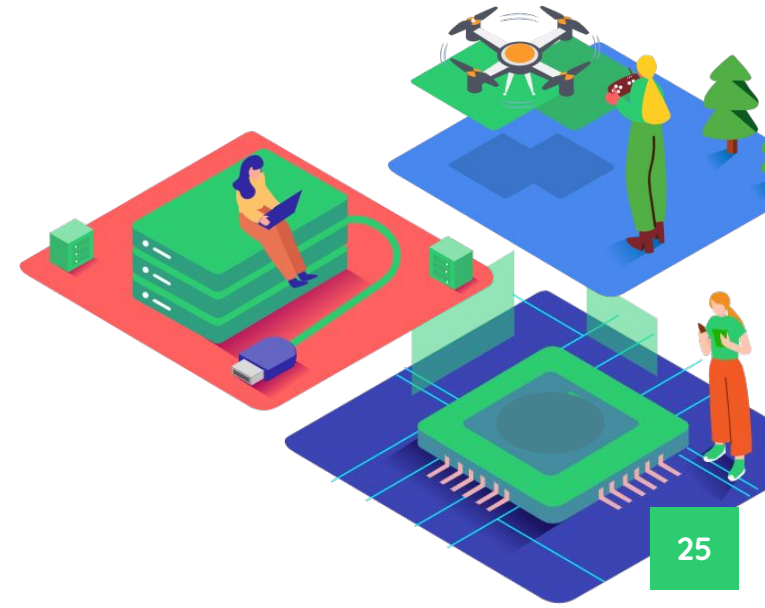
- ✓ The data is 'business' data: **large** and need to be updated.
- ✓ Consist of a lot of similar data (**homogeneous**)
- ✓ The data is relevant for **a long time**.
- ✓ **Simultaneous** usage by user.





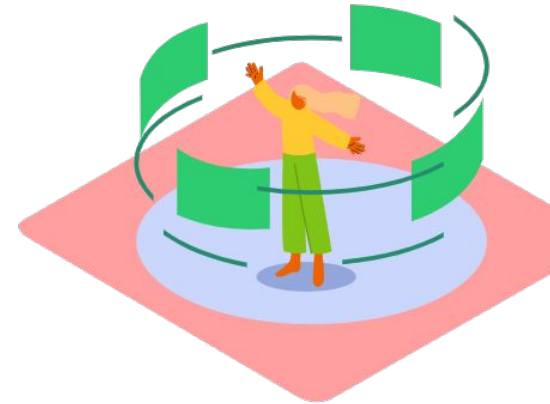
## When we don't need to use database?

- △ Simple, well-defined database applications **not expected to change at all**
- △ **Stringent**, real-time requirements that may not be met because of DBMS overhead
- △ Embedded systems with **limited storage capacity**
- △ **No multiple-user access** to data



## When databases cannot be used?

- ✗ If the database system **can not handle the complexity** of data because of the limitations of requirement modeling.
- ✗ If users **need special operations** which can not be met by the DBMS





## Inside a Google Data Center

<https://www.youtube.com/watch?v=XZmGGAbHqa0>

# Q&A

