Course Name:
Advanced Seminar on Conservation Medicine 2021

Relational Database

Kimihito Ito
Research Center for Zoonosis Control.

Points

- What a relational database is
- Normal forms of relational database
  - Examples of poor database design
- SQL

Relational database

- Relational database (RDB) is a database that is made up of a collection of relations
  - Microsoft Excel is NOT an RDB
- A relation is a table with columns and rows
- When you design a database, you need to think about relations that underly your data

How important RDB is

- RDB is running at the backend of information systems everywhere
  - Transactions on ATM machines of banks
  - Booking system of hotel rooms and train, airline, concert, and movie tickets
  - E commerce of online stores such as Amazon
  - Social networking system such as Twitter and Face book
  - Database of scientific papers
- Most information systems need an RDB

An example of relations

- A relation is a table with columns and rows.

Patients

Patient ID	**First name**	**Last name**	**Phone**
P0001	Jane	Doe	(555)555-1111
P0002	John	Doe	(555)555-2222
P0003	Jane	Smith	(555)555-3333
P0004	John	Smith	(555)555-4444

rows

columns

Properties of columns

- A name of column must be unique within table
  - No two columns have the same name
- A column must have values from the same type

Patients

Patient ID	**First name**	**Last name**	**Phone**
P0001	Jane	Doe	(555)555-1111
P0002	John	Doe	(555)555-2222
P0003	Jane	Smith	(555)555-3333
P0004	John	Smith	(555)555-4444

Properties of rows

- Only one value at the intersection of a column and row
  - A relation does not allow multivalued attributes such as a list
- There are no duplicate rows in a relation

Patients

Patient ID	First name	Last name	Phone
P0001	Jane	Doe	(555)555-1111
P0002	John	Doe	(555)555-2222
P0003	Jane	Smith	(555)555-3333
P0004	John	Smith	(555)555-4444

A primary key

- A primary key is a column or combination of columns that uniquely identifies each row
- A primary key is underlined

Patient ID	First name	Last name	Phone
P0001	Jane	Doe	(555)555-1111
P0002	John	Doe	(555)555-2222
P0003	Jane	Smith	(555)555-3333
P0004	John	Smith	(555)555-4444

A primary key

- A primary key is a column or combination of columns that uniquely identifies each row
- There are no duplicate rows in a relation.

⬇

Patients

Patient ID	First name	Last name	Phone
P0001	Jane	Doe	(555)555-1111
P0002	John	Doe	(555)555-2222
P0003	Jane	Smith	(555)555-3333
P0004	John	Smith	(555)555-4444

A primary key

- A primary key is a column or combination of columns that uniquely identifies each row.
- There are no duplicate rows in a relation.

Patients

Patient ID	First name	Last name	Phone
P0001	Jane	Doe	(555)555-1111
P0002	John	Doe	(555)555-2222
P0003	Jane	Smith	(555)555-3333
P0004	John	Smith	(555)555-4444

Requirements of primary keys

- A primary key should be some value that is highly unlikely ever to be null.
- A primary key should never change.

An ideal primary key!

Names may change. Phone numbers may be null.

Patient ID	**First name**	**Last name**	**Phone**
P0001	Jane	Doe	(555)555-1111
P0002	John	Doe	(555)555-2222
P0003	Jane	Smith	(555)555-3333
P0004	John	Smith	(555)555-4444

Concatenated primary keys

- Some tables have no single column in which the values never duplicate.
- Concatenated columns can be the primary key if each combination appear only once.

Order-lines

Order ID	Drug ID	Quantity
O0001	D0022	1
O0001	D0089	2
O0002	D0022	1
O0002	D1001	1

Concatenated primary keys

- Some tables have no single column in which the values never duplicate.
- Concatenated columns can be the primary key if each combination appear only once.

Cannot be a primary key due to duplicated values

⬇

Order ID	Drug ID	Quantity
O0001	D0022	1
O0001	D0089	2
O0002	D0022	1
O0002	D1001	1

Concatenated primary keys

- Some tables have no single column in which the values never duplicate.
- Concatenated columns can be the primary key if each combination appear only once.

Cannot be a primary key due to duplicated values

⬇

Order-lines

Order ID	Drug ID	Quantity
O0001	D0022	1
O0001	D0089	2
O0002	D0022	1
O0002	D1001	1

Concatenated primary keys

- Some tables have no single column in which the values never duplicate.
- Concatenated columns can be the primary key if each combination appear only once.

Cannot be a primary key due to duplicated values

Order-lines

Order ID	Drug ID	Quantity
O0001	D0022	1
O0001	D0089	2
O0002	D0022	1
O0002	D1001	1

Concatenated primary keys

- Some tables have no single column in which the values never duplicate.
- Concatenated columns can be the primary key if each combination appear only once.

No duplicated combinatios. They can be a concatenated primary key

Order ID	**Drug ID**	**Quantity**
O0001	D0022	1
O0001	D0089	2
O0002	D0022	1
O0002	D1001	1

Candidate key

- A column in concatenated columns is candidate key (or prime attribute) if the concatenated columns can be the primary key only with the column

Candidate key (prime attribute)

Candidate key (prime attribute)

Order ID	Drug ID	Quantity
O0001	D0022	1
O0001	D0089	2
O0002	D0022	1
O0002	D1001	1

Non-prime attribute

- A column is non-prime attribute if they it is not candidate key (or prime attribute)

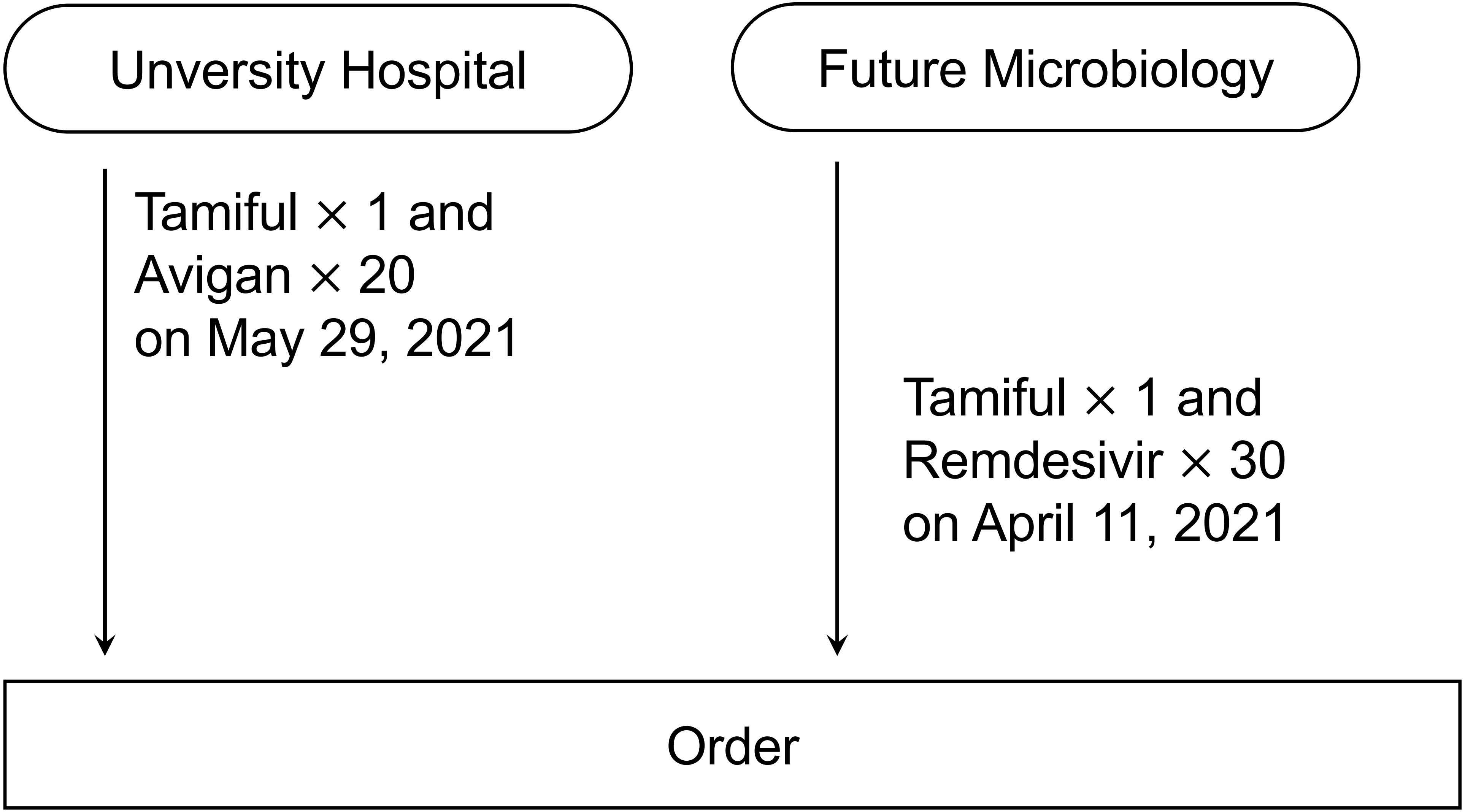
Candidate key (prime attribute)

Candidate key (prime attribute)

Non-prime attribute

Order ID	**Drug ID**	**Quantity**
O0001	D0022	1
O0001	D0089	2
O0002	D0022	1
O0002	D1001	1

An example of database design



Unversity Hospital

Future Microbiology

Tamiful × 1 and
Avigan × 20
on May 29, 2021

Tamiful × 1 and
Remdesivir × 30
on April 11, 2021

Order

An example of database design

Drugs

Drug ID	Name	Price
D0022	Tamiflu	$9.95
D0089	Avigan	$15.95
D1001	Remdesivir	$15.95

Customers

Custormer ID	Name	ZIP
C0186	University Hospital	060-0014
C1123	Bioinformatics Inc	001-0020
C3001	Future Microbiology	060-0018

Orders

Order ID	Customer ID	Order date
O0001	C3001	2021-03-29
O0002	C0186	2021-04-11

Order-lines

Order ID	Drug ID	Quantity	Shipped?
O0001	D0022	1	Y
O0001	D0089	20	Y
O0002	D0022	1	N
O0002	D1001	30	N

An example of database design

Drugs

Drug ID	Name	Price
D0022	Tamiflu	$9.95
D0089	Avigan	$15.95
D1001	Remdesivir	$19.95

Customers

Custormer ID	Name	ZIP
C0186	University Hospital	060-0014
C1123	Bioinformatics Inc	001-0020
C3001	Future Microbiology	060-0018

These table stores the information of “entities” such as drugs and customres. We can Identify entities by primary keys

An example of database design

These table stores the relationship between “entities” using primary keys of tables of the entities

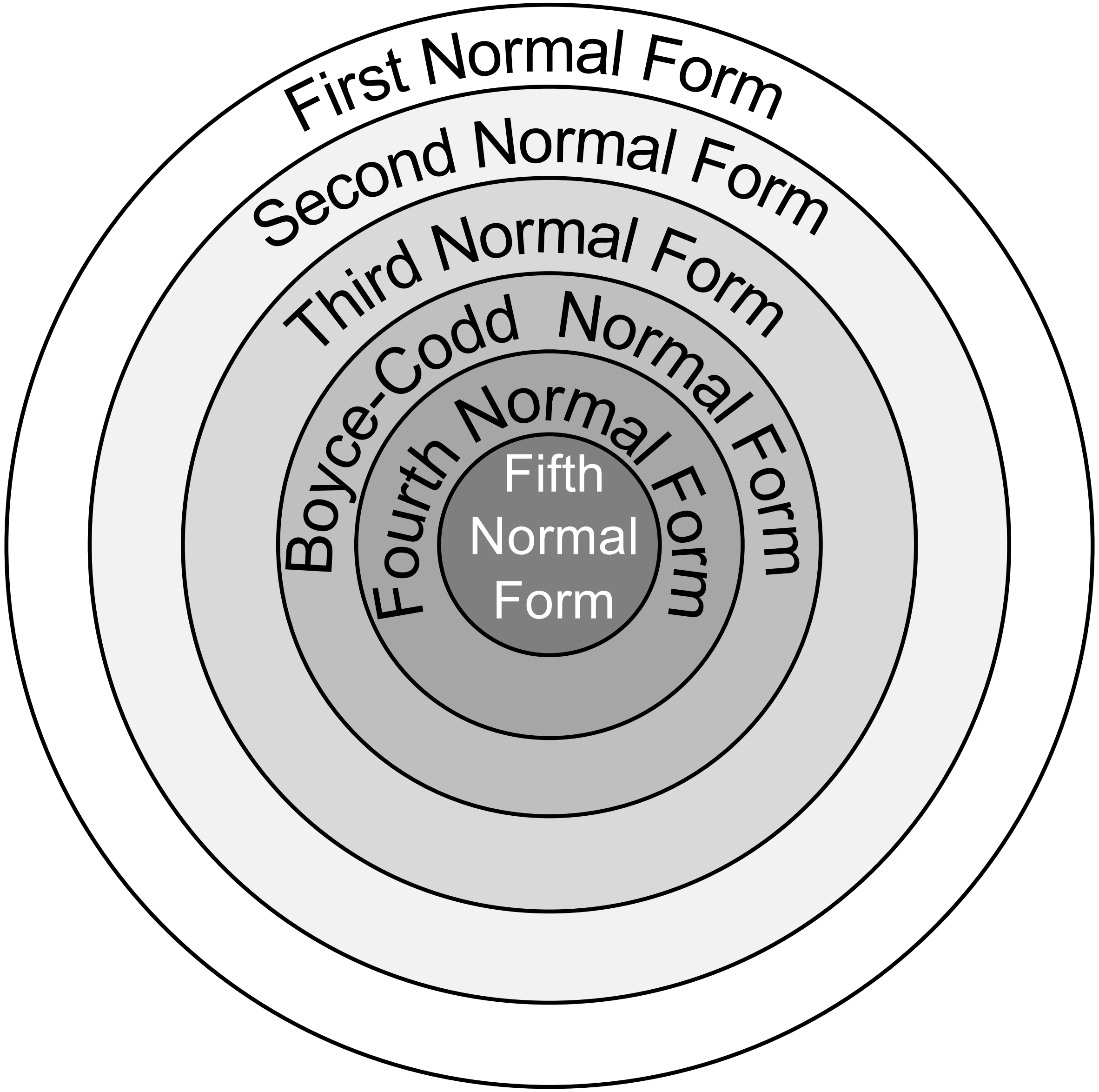
Orders

Order ID	Customer ID	Order date
O0001	C3001	2021-03-29
O0002	C0186	2021-04-11

Order-lines

Order ID	Drug ID	Quantity	Shipped?
O0001	D0022	1	Y
O0001	D0089	20	Y
O0002	D0022	1	N
O0002	D1001	30	N

Normal forms



First Normal form

- The data are stored in a two dimensional table with no repeating groups such as a list

Repeating groups of vaccines

⬇

Patient ID	First	Last	Vaccines	Type	Vaccination Dates
P0001	Jane	Doe	Pfizer	mRNA	2021-08-01
P0002	John	Doe	Modelna	mRNA	2021-07-24
P0003	Jane	Smith	Modelna	mRNA	2021-07-10, 2021-08-08
P0004	John	Smith	Sinovac, Pfizer	inactivated, mRNA	2021-04-01, 2021-07-05, 2021-07-26

First Normal form

- The data are stored in a two dimensional table with no repeating groups such as a list

Repeating groups of vaccination dates

Patient ID	**First**	**Last**	**Vaccines**	**Type**	**Vaccination Dates**
P0001	Jane	Doe	Pfizer	mRNA	2021-08-01
P0002	John	Doe	Modelna	mRNA	2021-07-24
P0003	Jane	Smith	Modelna	mRNA	2021-07-10, 2021-08-08
P0004	John	Smith	Sinovac, Pfizer	inactivated, mRNA	2021-04-01, 2021-07-05, 2021-07-26

Why repeating groups are bad

- Searching table is very difficult.
  - To know patients vaccinated before June 2021, individual dates need to be checked.
- There is no way to know which vaccine was used for each vaccination (0004).

Patient ID	First	Last	Vaccines	Type	Vaccination Dates
P0001	Jane	Doe	Pfizer	mRNA	2021-08-01
P0002	John	Doe	Modelna	mRNA	2021-07-24
P0003	Jane	Smith	Modelna	mRNA	2021-07-10, 2021-08-08
P0004	John	Smith	Sinovac, Pfizer	inactivated, mRNA	2021-04-01, 2021-07-05, 2021-07-26

Removing the repeating group

- Searching table get easier
  - Who is vaccinated before June 2021?
- Used vaccines were clarified (0004)

Patient ID	First	Last	Vaccine	Type	Date
P0001	Jane	Doe	Pfizer	mRNA	2021-08-01
P0002	John	Doe	Modelna	mRNA	2021-07-24
P0003	Jane	Smith	Modelna	mRNA	2021-07-10
P0003	Jane	Smith	Modelna	mRNA	2021-08-08
P0004	John	Smith	Sinovac	inactivated	2021-04-01
P0004	John	Smith	Pfizer	mRNA	2021-07-05
P0004	John	Smith	Pfizer	mRNA	2021-07-26

Problems with first normal form

- We need to update multiple records when the name of a patient changed (0003)
- No data is stored for unvaccinated patients

Patient ID	**First**	**Last**	**Vaccine**	**Type**	**Date**
P0001	Jane	Doe	Pfizer	mRNA	2021-08-01
P0002	John	Doe	Modelna	mRNA	2021-07-24
P0003	Jane	Smith	Modelna	mRNA	2021-07-10
P0003	Jane	Smith	Modelna	mRNA	2021-08-08
P0004	John	Smith	Sinovac	inactivated	2021-04-01
P0004	John	Smith	Pfizer	mRNA	2021-07-05
P0004	John	Smith	Pfizer	mRNA	2021-07-26

Functional dependency

- Attribute B is functionally dependent on Attribute A if for each unique value of A only one value of B is associated
  - Name is functionally dependent on Patient ID

Patient ID	First	Last	Vaccine	Type	Date
P0001	Jane	Doe	Pfizer	mRNA	2021-08-01
P0002	John	Doe	Modelna	mRNA	2021-07-24
P0003	Jane	Smith	Modelna	mRNA	2021-07-10
P0003	Jane	Smith	Modelna	mRNA	2021-08-08
P0004	John	Smith	Sinovac	inactivated	2021-04-01
P0004	John	Smith	Pfizer	mRNA	2021-07-05
P0004	John	Smith	Pfizer	mRNA	2021-07-26

Determinant

- Attribute B is functionally dependent on Attribute A if for each unique value of A only one value of B is associated
  - Attribute A is called determinant of Attribute B

Patient ID	**First**	**Last**	**Vaccine**	**Type**	**Date**
P0001	Jane	Doe	Pfizer	mRNA	2021-08-01
P0002	John	Doe	Modelna	mRNA	2021-07-24
P0003	Jane	Smith	Modelna	mRNA	2021-07-10
P0003	Jane	Smith	Modelna	mRNA	2021-08-08
P0004	John	Smith	Sinovac	inactivated	2021-04-01
P0004	John	Smith	Pfizer	mRNA	2021-07-05
P0004	John	Smith	Pfizer	mRNA	2021-07-26

Second Normal form

- The relation is in first normal form
- No non-prime attribute functionally dependent on a part of a candidate key
  - Table below is NOT second normal form because First and Last are non-prime attribute and functionally dependent upon a candidate key, Patient ID

Patient ID	**First**	**Last**	**Vaccine**	**Type**	**Date**
P0001	Jane	Doe	Pfizer	mRNA	2021-08-01
P0002	John	Doe	Modelna	mRNA	2021-07-24
P0003	Jane	Smith	Modelna	mRNA	2021-07-10
P0003	Jane	Smith	Modelna	mRNA	2021-08-08
P0004	John	Smith	Sinovac	inactivated	2021-04-01
P0004	John	Smith	Pfizer	mRNA	2021-07-05
P0004	John	Smith	Pfizer	mRNA	2021-07-26

Second Normal form

- We don’t have to update multiple records when the name of a patient changed
- We can store data on unvaccinated patients

Patients

Patient ID	**First**	**Last**
P0001	Jane	Doe
P0002	John	Doe
P0003	Jane	Smith
P0004	John	Smith
P0005	Paul	Smith

Vaccination

Patient ID	**Vaccine**	**Type**	**Date**
P0001	Pfizer	mRNA	2021-08-01
P0002	Modelna	mRNA	2021-07-24
P0003	Modelna	mRNA	2021-07-10
P0003	Modelna	mRNA	2021-08-08
P0004	Sinovac	inactivated	2021-04-01
P0004	Pfizer	mRNA	2021-07-05
P0004	Pfizer	mRNA	2021-07-26

Third Normal form

- The relation is second normal form
- All columns are functionaly dependent on sololy on the primary key

Third normal form

Patients

Patient ID	**First**	**Last**
P0001	Jane	Doe
P0002	John	Doe
P0003	Jane	Smith
P0004	John	Smith

Not third normal form

Vaccination

Patient ID	**Vaccine**	**Type**	**Date**
P0001	Pfizer	mRNA	2021-08-01
P0002	Modelna	mRNA	2021-07-24
P0003	Modelna	mRNA	2021-07-10
P0003	Modelna	mRNA	2021-08-08
P0004	Sinovac	inactivated	2021-04-01
P0004	Pfizer	mRNA	2021-07-05
P0004	Pfizer	mRNA	2021-07-26

Third Normal form

Patients

Patient ID	**First**	**Last**
P0001	Jane	Doe
P0002	John	Doe
P0003	Jane	Smith
P0004	John	Smith

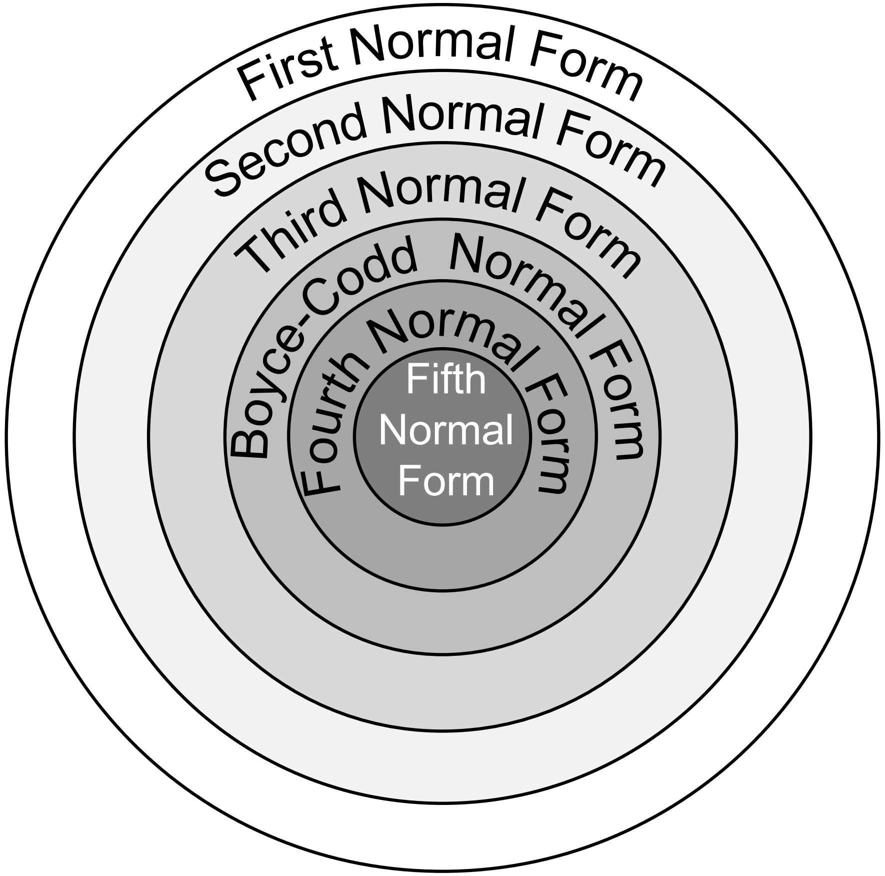
Vaccines

Vaccine ID	**Manufacturer**	**Type**
V0001	Pfizer	mRNA
V0002	Modelna	mRNA
V0003	Sinovac	inactivated

Vaccination

Patient ID	**Vaccine ID**	**Date**
P0001	V0001	2021-08-01
P0002	V0002	2021-07-24
P0003	V0002	2021-07-10
P0003	V0002	2021-08-08
P0004	V0003	2021-04-01
P0004	V0001	2021-07-05
P0004	V0001	2021-07-26

Normal forms



- For most relations, third normal form is a good design objective.
- Relations in third nomal form are free of most anomalies.
- Please refer textbooks for higher normal forms

SQL

- Structured English Query Language (SEQUEL; SQL) is a computer language that has been implemented in the most relational database management system (DBMS).
- SQL was developed by IBM in the early 1970s.
- SQL can be used to create and update tables, and to retrieve information from tables.

SQL

- SQL is used to manage RDB running at the backend of information system
- You don't need to write SQL codes
- Computers can generate SQL codes from your clicks on the Web browsers
  - When you reserve a hotel room at a web site, its server generates an SQL code and shows you the results on your browser
- Knowing SQL is helpful to design an information system using RDB

SQL SELECT FROM

- SELECT FROM statement retrieve data in columns from tables
  - SELECT columns
    FROM table

Example

Patients

Patient_ID	**First**	**Last**
P0001	Jane	Doe
P0002	John	Doe
P0003	Jane	Smith
P0004	John	Smith

Vaccines

Vaccine_ID	**Manufacturer**	**Type**
V0001	Pfizer	mRNA
V0002	Modelna	mRNA
V0003	Sinovac	inactivated

Vaccination

Patient_ID	**Vaccine_ID**	**Date**
P0001	V0001	2021-08-01
P0002	V0002	2021-07-24
P0003	V0002	2021-07-10
P0003	V0002	2021-08-08
P0004	V0003	2021-04-01
P0004	V0001	2021-07-05
P0004	V0001	2021-07-26

Example

```
SELECT Manufacturer FROM Vaccines;

Manufacturer
------------
Pfizer
Modelna
Sinovac
```

Vaccines

Vaccine_ID	Manufacturer	Type
V0001	Pfizer	mRNA
V0002	Modelna	mRNA
V0003	Sinovac	inactivated

Example

```
SELECT Manufacturer,Type FROM Vaccines;

Manufacturer | Type
----------------------------
Pfizer       | mRNA
Modelna      | mRNA
Sinovac      | inactivated
```

Vaccines

Vaccine_ID	Manufacturer	Type
V0001	Pfizer	mRNA
V0002	Modelna	mRNA
V0003	Sinovac	inactivated

SQL WHERE

- WHERE clause retrieves rows conditioning with predicates
  - SELECT columns
    FROM table
    WHERE predicate
- You can use the following in predicates
  - relationship operators e.g., ‘=’, ‘>’, and ‘<’.
  - logical operators e.g., AND, OR, and NOT
  - other special operators e.g., IN and LIKE

Example

```
SELECT Manufacturer FROM Vaccines
WHERE Type=‘mRNA’;

Manufacturer
------------
Pfizer
Modelna
```

Vaccines

Vaccine_ID	Manufacturer	Type
V0001	Pfizer	mRNA
V0002	Modelna	mRNA
V0003	Sinovac	inactivated

Example

```
SELECT Patient_ID FROM Vacctination
WHERE Date>‘2021-07-31’;

Patient_ID
----------
P0001
P0003
```

Vaccination

Patient_ID	**Vaccine_ID**	**Date**
P0001	V0001	2021-08-01
P0002	V0002	2021-07-24
P0003	V0002	2021-07-10
P0003	V0002	2021-08-08
P0004	V0003	2021-04-01
P0004	V0001	2021-07-05
P0004	V0001	2021-07-26

Retrieval from multiple tables

- List the tables to be combined after FROM to retrieve data from combined tables.

  -
    ```
    SELECT columns
    FROM table1, table2
    WHERE table1.column_a = table2.column_b
    ```

Example

```
SELECT First, Last FROM Patients, Vaccination
WHERE Vaccination.Date > ‘2021-07-31’ AND
Vaccination.Patient_ID = Patient.Patient_ID;

First | Last
-------------
Jane  | Doe
Jane  | Smith
```

Vaccination

<u>**Patient_ID**</u>	**Vaccine_ID**	<u>**Date**</u>
P0001	V0001	2021-08-01
P0002	V0002	2021-07-24
P0003	V0002	2021-07-10
P0003	V0002	2021-08-08
P0004	V0003	2021-04-01
P0004	V0001	2021-07-05
P0004	V0001	2021-07-26

Patients

<u>**Patient_ID**</u>	**First**	**Last**
P0001	Jane	Doe
P0002	John	Doe
P0003	Jane	Smith
P0004	John	Smith

Practice

Patients

Patient_ID	**First**	**Last**
P0001	Jane	Doe
P0002	John	Doe
P0003	Jane	Smith
P0004	John	Smith

Vaccines

Vaccine_ID	**Manufacturer**	**Type**
V0001	Pfizer	mRNA
V0002	Modelna	mRNA
V0003	Sinovac	inactivated

Vaccination

Patient_ID	**Vaccine_ID**	**Date**
P0001	V0001	2021-08-01
P0002	V0002	2021-07-24
P0003	V0002	2021-07-10
P0003	V0002	2021-08-08
P0004	V0003	2021-04-01
P0004	V0001	2021-07-05
P0004	V0001	2021-07-26

Write an SQL code that looks for patients who got a shot of inactivated vaccine

Example

```
SELECT First, Last
FROM Patients, Vaccination, Vaccines
WHERE Vaccines.Type=‘inactivated’ AND
Vaccination.Vaccine_ID=Vaccines.Vaccine_ID AND
Vaccination.Patient_ID = Patient.Patient_ID;

First | Last
-----------
John  | Smith
```

SQL UPDATE

- You can modify information in the database by UPDATE command in SQL

```
UPDATE Patients
SET last=‘Yamada’
WHERE Patient_ID=‘P0003’
```

Patients

Patient ID	**First**	**Last**
P0001	Jane	Doe
P0002	John	Doe
P0003	Jane	Yamada
P0004	John	Smith

SQL UPDATE and INSERT

- You can add information in the database by INSERT command in SQL

```
INSERT INTO Patients
VALUES (‘P0005’,‘Paul’,‘Smith’);
```

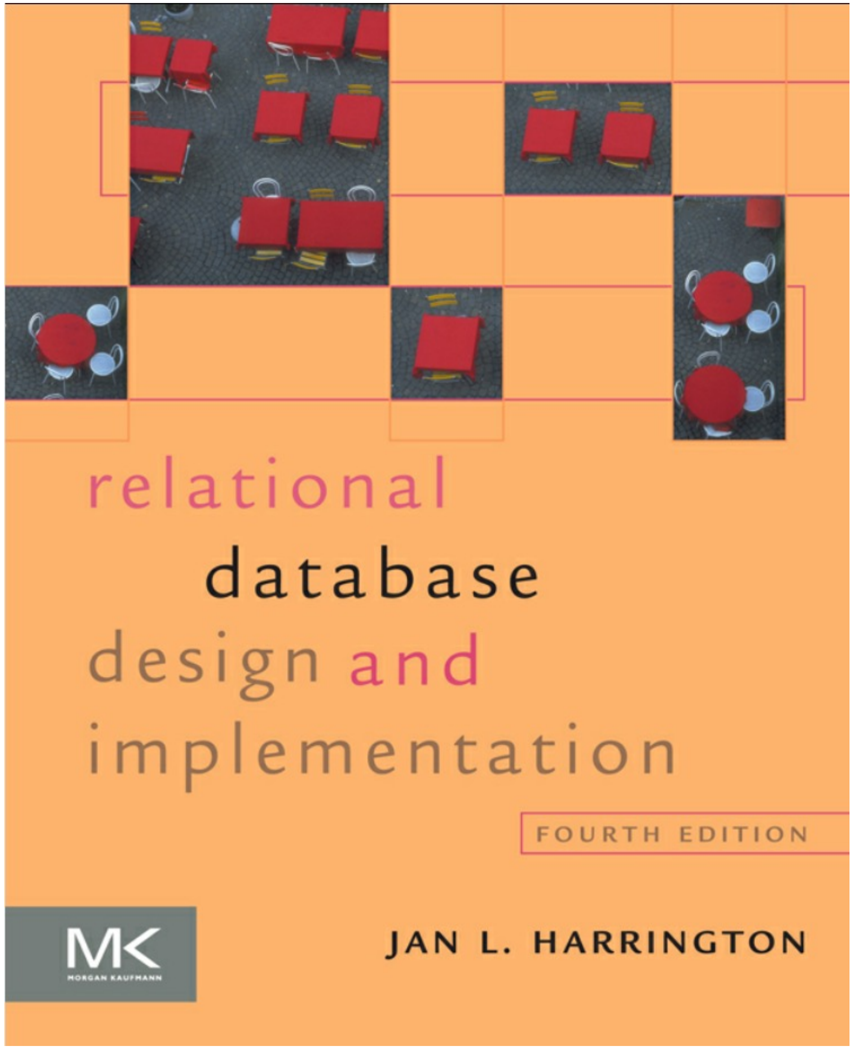
Patients

Patient ID	First	Last
P0001	Jane	Doe
P0002	John	Doe
P0003	Jane	Smith
P0004	John	Smith
P0005	Paul	Smith

Points

- What a relational database is
- Normal Forms of Relational Database
  - Examples of poor database design
- SQL

Textbook



Harrington, Jan L. *Relational Database Design and Implementation,* Morgan Kaufmann

Appendix

Boyce–Codd normal form and

Fourth normal form

Boyce–Codd Normal form

- The relation is in third normal form
- All determinants are candidate keys

Not Boyce–Codd normal form

Reservations

Date	**Room**	**Price**
2021-07-01	small	$100
2021-07-07	large	$200
2021-07-07	small	$200
2021-07-24	small	$100
2021-08-04	large	$400
2021-08-30	large	$400

Boyce–Codd Normal form

- The relation is in third normal form
- All determinants are candidate keys

Boyce–Codd normal form

Prices

Room	**Membership**	**Price**
small	yes	$100
small	no	$200
large	yes	$200
large	no	$400

Boyce–Codd normal form

Reservations

Date	**Room**	**Membership**
2021-07-01	small	yes
2021-07-07	large	yes
2021-07-07	small	no
2021-07-24	small	yes
2021-08-04	large	no
2021-08-30	large	no

Fourth Normal form

- The relation is in Boyce-Codd normal form
- There are no multivalued dependencies

Not fourth normal form

Paper ID	Author	Reference
P0003	A0001	P0001
P0003	A0001	P0002
P0003	A0002	P0001
P0003	A0002	P0002

Fourth Normal form

- The relation is in Boyce-Codd normal form
- There are no multivalued dependencies

Fourth normal form

Authors

Paper ID	**Author**
P0003	A0001
P0003	A0002

Fourth normal form

References

Paper ID	**Reference**
P0003	P0001
P0003	P0002