# Chapter 2. Introduction to the Relational Model

## 2.1) Structure of Relational Databases.

Remark. RDB consists of a collection of tables with unique names.
Row in a table represents a relationship among a set of values.

| Mathematics | | Database |
|---|---|---|
| Relation | $\longleftrightarrow$ | Table. |
| Tuple | $\longleftrightarrow$ | Row. |
| Attribute | $\longleftrightarrow$ | Column. |

Def. Relation instance is a specific instance of relation,
i.e., containing a specific set of tuples.

Remark. Values of attributes of a tuple must be uniquely identifiable,
i.e., no two rows are the same in a table.

Remark. Order of tuples is irrelevant.

Def. Domain of the attribute is a set of permitted values.
For all relation $r$, the domain of all the attributes in $r$ is atomic,
i.e., not consisted of several values, so elements are indivisible.

Def. The null value is a special value that signifies that
the value is unknown or does not exist.

Remark. Null values causes difficulties when we access or update databases,
thus should be eliminated if possible.

## (2.2) Database Schema.

Def. **Database schema** is the logical design of the database.

**Database instance** is a snapshot of the data in database given time.

## (2.3) Keys.

Def. Given a relation $r$, set of attributes in the schema of relation $r$ $R$,

tuple $t_i, t_j \in r$ and $i \neq j$ ( so that $t_i \neq t_j$ ).

A **superkey** $K$ of $r$ is $K \subseteq R$ such that $t_i.K \neq t_j.K$

i.e., no two distinct tuples have the same values on all attributes in $K$.

Def. A **candidate key** $C$ is a superkey of $r$

such that no subset except itself is a superkey.

i.e., a minimal superkey.

Def. A **primary key** is a candidate key chosen by the designer

as the principle means of identifying a tuple in a relation.

Def. Given two distinct relation $r_1, r_2$,

if $r_1$ include the primary key $K$ of $r_2$ among its attributes,

$K$ is called a **foreign key** from $r_1$ referencing $r_2$,

$r_1$ is the **referencing relation** of the foreign key dependency,

and $r_2$ is the **referenced relation** of the foreign key.

Remark. **Referential integrity constraint** requires that

the values appeared as the foreign key from $r_1$ referencing $r_2$

has to appear in at least one tuple $t \in r_2$.

## (2.4) Schema Diagrams.



Referencing relation.

Referenced relation.

primary key.

Foreign key dependencies
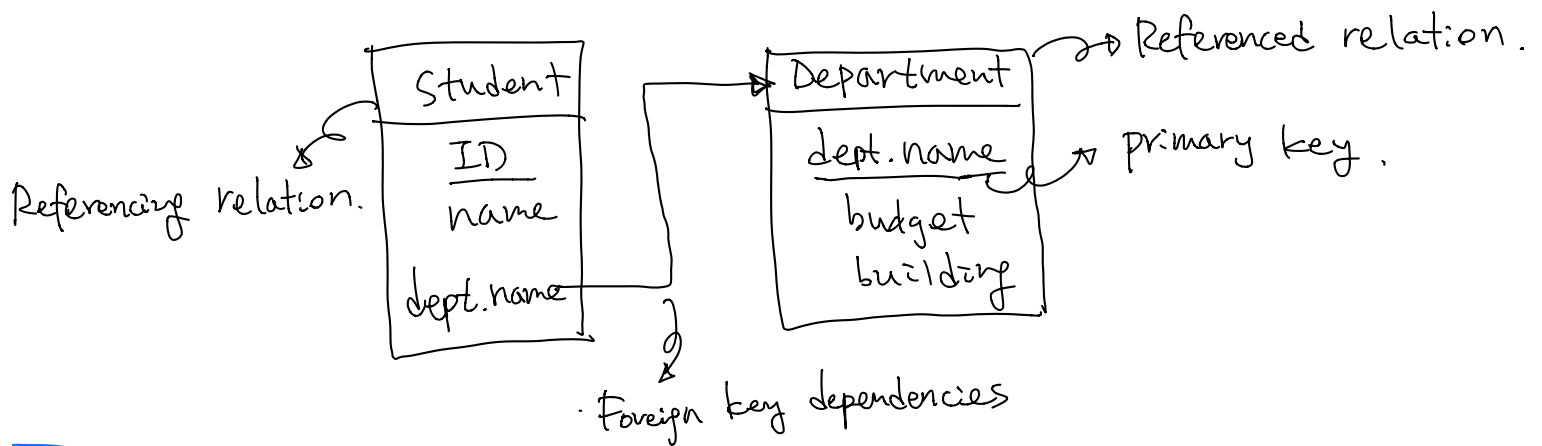
## (2.5) Relational Query Languages

<u>Def</u>. A procedural language describes a sequence of operations on the database to acquire desired results.

<u>Example</u>. Relational algebra.

<u>Def</u>. A nonprocedural language describes the desired result itself.

<u>Example</u>. Tuple relational calculus. Domain relational calculus.

<u>Remark</u>. Query languages in practice include elements of both.

## (2.6) Relational Operators.

: Takes one or more relations as input and outputs a relation.

i) $\sigma$ (selection) : Tuples which satisfy given predicate. ex) $\sigma_{attr > 0}$ (relation)

ii) $\Pi$ (projection) : Tuples with specified attributes. ex) $\Pi_{attr_1, attr_2}$ (relation)

iii) $\bowtie$ (Natural join) : Tuples from two relations that have the same attribute values. ex) $r_1 \bowtie r_2$.

iv) $\times$ (Cartesian product) : Every tuple from two relations (Regardless of whether values match or not). ex) $r_1 \times r_2$.

v) $\cup$ (Union) : Union of tuples from two relations. ex) $\Pi_{attr_1}(r_1) \cup \Pi_{attr_2}(r_2)$