

COMPS320F Database Management

Lecture 2

2021.9.27 Jiawei Wang (16:00 – 18:00)

THE RELATIONAL MODEL

Terri Wong

School of Science and Technology

Content

Terminology of relational model.

How tables are used to represent data.

Connection between mathematical relations and relations in the relational model.

Properties of database relations.

How to identify Candidate keys (CK), Primary keys (PK), and Foreign keys (FK).

Meaning of entity integrity and referential integrity.

Purpose and advantages of views.

Relational Model Terminology

A relation is a table with columns and rows.

- Only applies to logical structure of the database, not the physical structure.

Attribute is a named column of a relation.

Domain is the set of allowable values for one or more attributes.

Relational Model Terminology

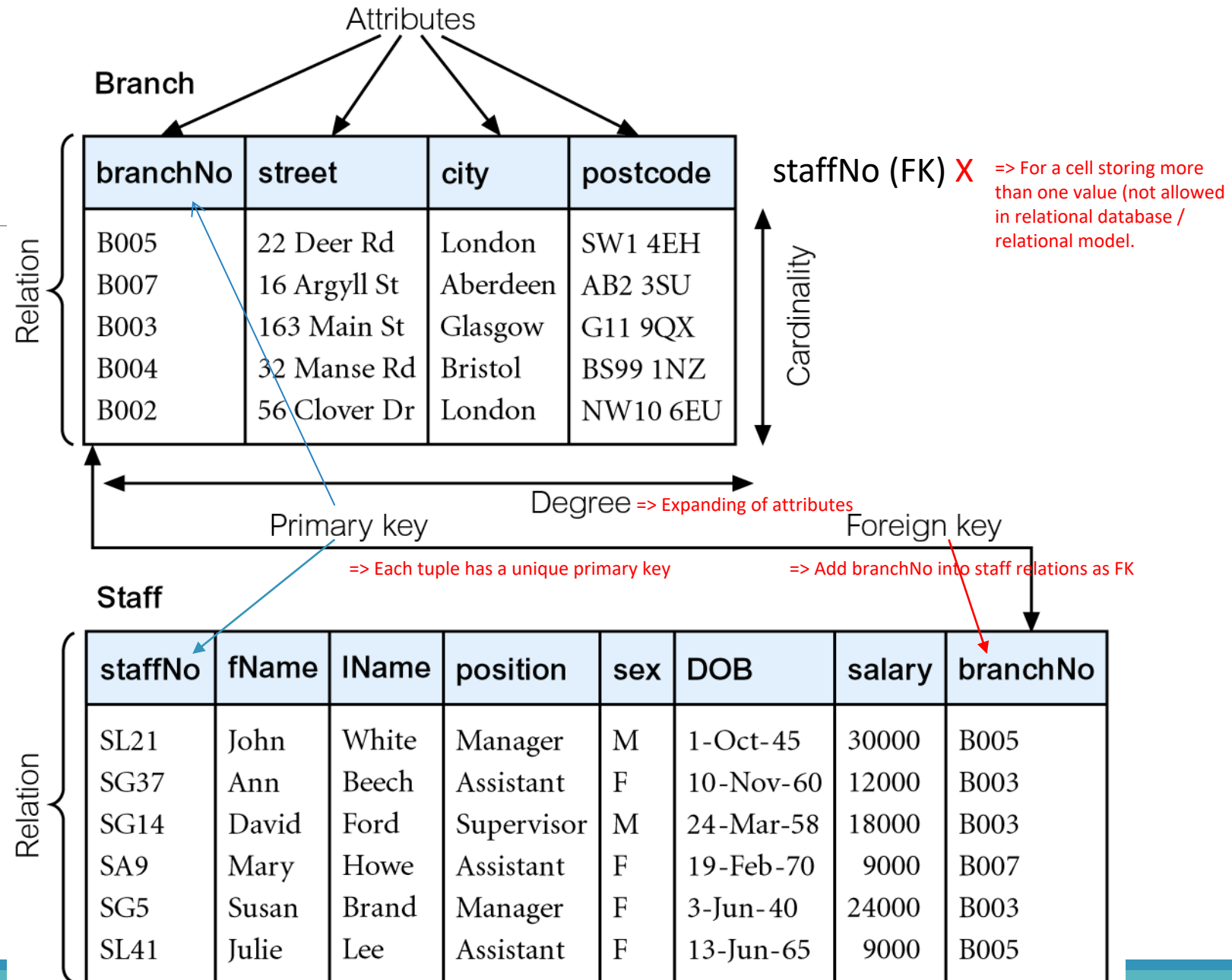
Tuple is a row of a relation.

Degree is the number of attributes in a relation.

Cardinality is the number of tuples in a relation.

Relational Database is a collection of normalized relations with distinct relation names. (Duplicated relation names not allowed)

Instances of Branch and Staff Relations



Examples of Attribute Domains => How do we represent the attributes

Attribute	Domain Name	Meaning	Domain Definition
branchNo	BranchNumbers	The set of all possible branch numbers	character: size 4, range B001–B999
street	StreetNames	The set of all street names in Britain	character: size 25
city	CityNames	The set of all city names in Britain	character: size 15
postcode	Postcodes	The set of all postcodes in Britain	character: size 8
sex	Sex	The sex of a person	character: size 1, value M or F
DOB	DatesOfBirth	Possible values of staff birth dates	date, range from 1-Jan-20, format dd-mmm-yy
salary	Salaries	Possible values of staff salaries	monetary: 7 digits, range 6000.00–40000.00

Alternative Terminology for Relational Model (别名)

Formal terms	Alternative 1	Alternative 2
Relation	Table	File
Tuple	Row	Record
Attribute	Column	Field

Database Relations

Relation schema

- Named relation defined by a set of attribute and domain name pairs.

Relational database schema

- Set of relation schemas, each with a distinct name.

University (database schema) => relation schema i.e., student, staff, course, etc.

Properties of Relations (Table)

Relation name is distinct from all other relation names in relational schema.

Each cell of relation contains exactly one atomic (single) value.

Each attribute has a distinct name.

Values of an attribute are all from the same **domain**.

=> i.e., HKMU only stores students details from HKMU only

Properties of Relations

Each tuple is distinct; there are no duplicate tuples.

Order of attributes has no significance.

Order of tuples has no significance, theoretically.

Relational Keys

Superkey

- An attribute, or set of attributes, that uniquely identifies a tuple within a relation.
- The set of attributes which can uniquely identify rows of the employee table, which are:

{Emp_HKID}

{Emp_No}

{Emp_HKID, Emp_No}

{Emp_HKID, Emp_Name}

{Emp_HKID, Emp_No, Emp_Name}

{Emp_No, Emp_Name}

Employee

Emp_HKID	Emp_No	Emp_Name
T223344	E001	Ethan Lee
Y004488	E002	Jacob Chan
H007733	E003	Evelyn Wong
K577722	E004	Ethan Lee
V487766	E005	David Leung

Relational Keys

Candidate Key

- Superkey (K) such that no proper subset is a superkey within the relation.
- In each tuple of R, values of K uniquely identify that tuple (uniqueness).
- No proper subset of K has the uniqueness property (irreducibility).
- The **minimal superkeys with no redundant attributes**, which are:

{Emp_HKID}

{Emp_No}

- [Note]

Only these two sets are candidate keys as all other sets are having redundant attributes that are not necessary for unique identification.

In the above example, we have not chosen {Emp_HKID, Emp_Name} as candidate key because {Emp_HKID} alone can identify a unique row in the table and Emp_Name is redundant.

Employee

Emp_HKID	Emp_No	Emp_Name
T223344	E001	Ethan Lee
Y004488	E002	Jacob Chan
H007733	E003	Evelyn Wong
K577722	E004	Ethan Lee
V487766	E005	David Leung

Relational Keys

This semester: Only one to many or one to one relationship

Primary Key

- Candidate key selected to identify tuples uniquely within relation.
- So Either {Emp_HKID} or {Emp_No} can be the primary key.
- Usually {Emp_No} will be selected as primary key which is more convenient to use within the company.

{Emp_No}

Alternate Keys

- Candidate keys that are not selected to be primary key.

{Emp_HKID}

- *[Note] - If the table only contains ONE candidate key which has already selected as primary key, leaving no other candidate key left, in this case there will be no alternate key for such table.*

Employee

Emp_HKID	Emp_No	Emp_Name
T223344	E001	Ethan Lee
Y004488	E002	Jacob Chan
H007733	E003	Evelyn Wong
K577722	E004	Ethan Lee
V487766	E005	David Leung

Relational Keys

Foreign Key

- Attribute, or set of attributes, within one relation that matches candidate key of some (possibly same) relation.

For table Dept:

Primary key:

- {DeptID}

Foreign Key:

- Nil

For table Employee:

Primary key:

- {Emp_No}

Foreign Key:

- The {DeptID} in Employee table is the foreign key references Dept table.

Dept

<u>DeptID</u>	DeptName	Location
D001	Finance	Hong Kong
D002	Sales	Japan
D003	Sales	Hong Kong
D004	Human Resources	Japan

Employee

<u>Emp_No</u>	Emp_Name	<i>DeptID</i>
E001	Ethan Lee	<i>D001</i>
E002	Jacob Chan	<i>D002</i>
E003	Evelyn Wong	<i>D002</i>
E004	Ethan Lee	<i>D001</i>
E005	David Leung	<i>D002</i>

Relationship between the keys

Super keys
(may not be minimal)

=> Most possible and unique search keys

Candidate keys (minimal)

Primary key [selected one]

Alternate keys

Foreign keys
= selected PK(s) of other tables

Integrity Constraints

Empty values in cells

Null

- Represents value for an attribute that is currently unknown or not applicable for tuple.
- Deals with incomplete or exceptional data.
- Represents the absence of a value and is not the same as zero or spaces, which are values.

Employee

Emp_HKID	Emp_No	Emp_Name	Remark
T223344	E001	Ethan Lee	Top Customer Rating
Y004488	E002	Jacob Chan	
H007733	E003	Evelyn Wong	Top Sales
K577722	E004	Ethan Lee	NULL
V487766	E005	David Leung	Excellent Award

Both are NULL values

Integrity Constraints

Entity Integrity

- In a base relation, no attribute of a primary key can be null.

Referential Integrity

- If foreign key exists in a relation, foreign key value **must match a candidate** key value of some tuple in its home relation.

Dept

<u>DeptID</u>	DeptName	Location
D001	Finance	Hong Kong
D002	Sales	Japan
D003	Sales	Hong Kong
D004	Human Resources	Japan

Employee

<u>Emp_No</u>	Emp_Name	<i>DeptID</i>
E001	Ethan Lee	<i>D001</i>
E002	Jacob Chan	<i>D002</i>
E003	Evelyn Wong	<i>D002</i>
E004	Ethan Lee	<i>D001</i>
E005	David Leung	<i>D002</i>

Integrity Constraints

General Constraints

- Additional rules specified by users or database administrators that define or constrain some aspect of the enterprise.
- i.e. the length of Emp_No consists exactly 5 characters starting with capital letter E.

Employee

Emp_HKID	Emp_No	Emp_Name
T223344	E001	Ethan Lee
Y004488	E002	Jacob Chan
H007733	E003	Evelyn Wong
K577722	E004	Ethan Lee
V487766	E005	David Leung

Views

Base Relation

- Named relation corresponding to an entity in conceptual schema, whose tuples are physically stored in database.

View

- Dynamic result of one or more relational operations operating on base relations to produce another relation.

Views

A virtual relation that does not necessarily actually exist in the database but is produced upon request, at time of request.

Contents of a view are defined as a query on one or more base relations.

Views are **dynamic**, meaning that changes made to base relations that affect view attributes are immediately reflected in the view.

i.e. a view may be created for employees from Hong Kong to retrieve sales and finance data of Hong Kong region only.

Dept

<u>DeptID</u>	DeptName	Location
D001	Finance	Hong Kong
D002	Sales	Japan
D003	Sales	Hong Kong
D004	Human Resources	Japan

Employee

<u>Emp_No</u>	Emp_Name	DeptID
E001	Ethan Lee	D001
E002	Jacob Chan	D002
E003	Evelyn Wong	D002
E004	Ethan Lee	D001
E005	David Leung	D002
E006	Rachel Cheung	D003

Purpose of Views

Provides powerful and flexible security mechanism by hiding parts of database from certain users.

Permits users to access data in a customized way, so that same data can be seen by different users in different ways, at same time.

Can simplify complex operations on base relations.

Updating Views

All updates to a base relation should be immediately **reflected in all views** that reference that base relation.

If view is updated, underlying base relation should reflect change.

Reference

Chapters 4 of Connolly, T and Begg, C (2015), Database Systems: A practical Approach to Design, Implementation, and Management (6th ed.), Boston: Pearson Education.