Chapter 4

The Relational Model

Chapter 4 - Objectives

- The terminology of the relational model
- How tables are used to represent data
- The connection between mathematical relations and relations in the relational model
- Properties of database relations
- How to identify candidate, primary, alternate, and foreign keys
- The meaning of entity integrity and referential integrity
- The purpose and advantages of views in relational systems

Relational Model Terminology

- Relational model is based on math concept of a relation physically represented as a table
- RDBMS requires the database be perceived as a table with columns and rows.
  - Only applies to logical structure of the database, not the physical structure.
- The rows correspond to individual records
- The columns correspond to attributes and can be in any order
- Domain is the set of allowable values for one or more attributes.

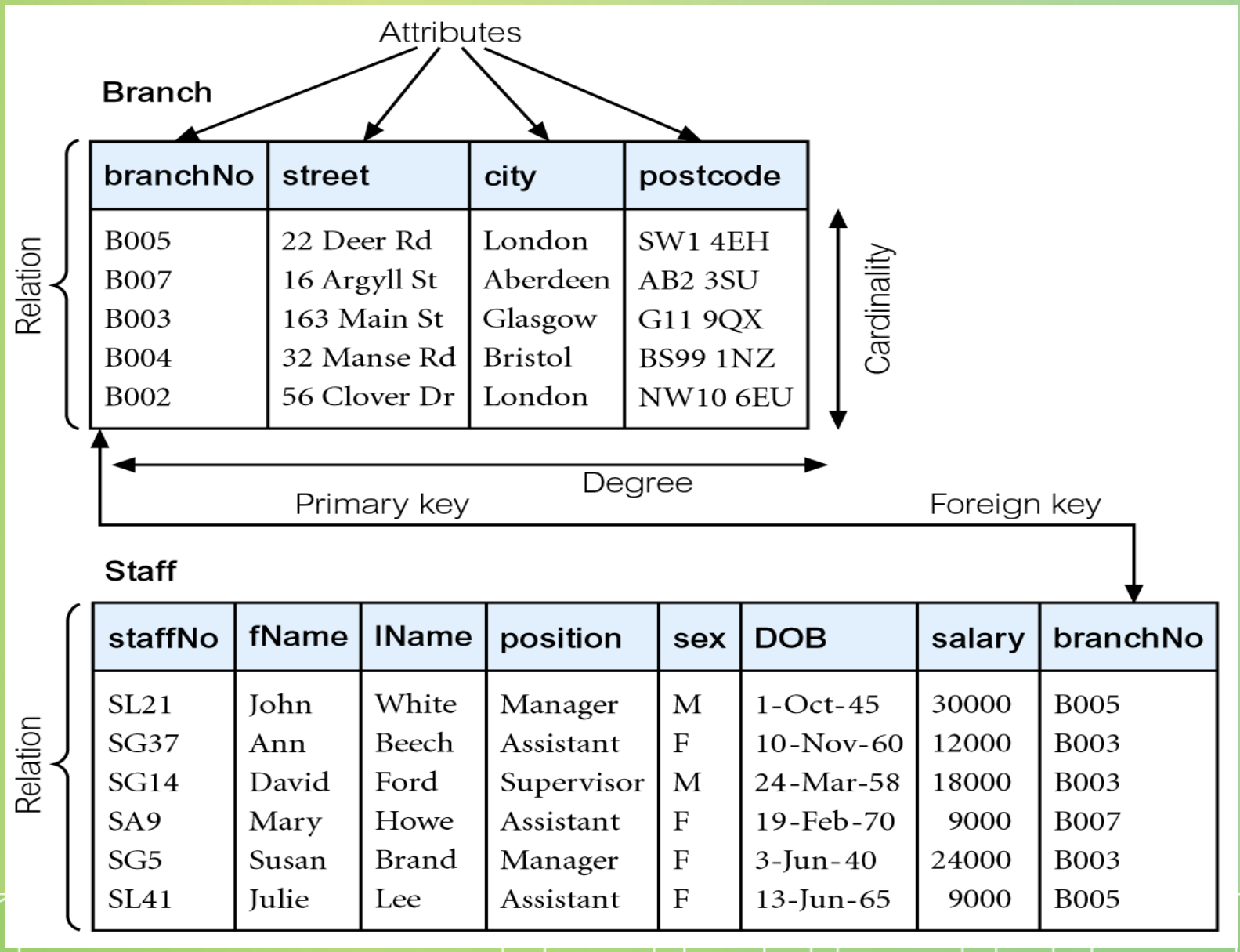
Relational Model Terminology

- 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.
- Normalized relations have no repeating groups – each cell contains only one value

Relational Model Terminology

- Intension is the structure of a relation, together with a specification of the domains and any other restrictions on possible values. The intension is usually fixed.
- The extension is an instance of the tuples of a relation. The extension is dynamic, changing frequently due to insertions, deletions, and updates to the data.

Instances of Branch and Staff Relations



Branch

branchNo	street	city	postcode
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU

Staff

staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

Examples of Attribute Domains

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

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

Mathematical Definition of Relation

- Consider two sets, D_1 & D_2, where $D_1 = \{2, 4\}$ and $D_2 = \{1, 3, 5\}$.
- Cartesian product, $D_1 \times D_2$, is set of all ordered pairs, where first element is member of D_1 and second element is member of D_2.

$$D_1 \times D_2 = \{(2, 1), (2, 3), (2, 5), (4, 1), (4, 3), (4, 5)\}$$

- Alternative way is to find all combinations of elements with first from D_1 and second from D_2.

Mathematical Definition of Relation

- Any subset of Cartesian product is a relation; e.g.

  $R = \{(2, 1), (4, 1)\}$

- May specify which pairs are in relation using some condition for selection; e.g.
  - second element is 1:

    $R = \{(x, y) \mid x \in D_1, y \in D_2, \text{ and } y = 1\}$
  - first element is always twice the second:

    $S = \{(x, y) \mid x \in D_1, y \in D_2, \text{ and } x = 2y\}$

Mathematical Definition of Relation

- Consider three sets D_1, D_2, D_3 with Cartesian Product $D_1 \times D_2 \times D_3$; e.g.

$D_1 = \{1, 3\}$ $D_2 = \{2, 4\}$ $D_3 = \{5, 6\}$

$D_1 \times D_2 \times D_3 = \{(1,2,5), (1,2,6), (1,4,5), (1,4,6), (3,2,5), (3,2,6), (3,4,5), (3,4,6)\}$

- Any subset of these ordered triples is a relation.

Mathematical Definition of Relation

- **Cartesian product of n sets $(D_1, D_2, \ldots, D_n)$ is:**

$$D_1 \times D_2 \times \ldots \times D_n = \{(d_1, d_2, \ldots, d_n) \mid d_1 \in D_1, d_2 \in D_2, \ldots, d_n \in D_n\}$$

usually written as:

$$\prod_{i=1}^{n} D_i$$

- **Any set of n-tuples from this Cartesian product is a relation on the n sets.**

Database Relations

- Relation schema
  - Let A_1, A_2, …, A_n be attributes with domains D_1, D_2,…, D_n. Then the set $\{A_1:D_1, A_2:D_2, \ldots A_n:D_n\}$ is a relation schema.
- Relational database schema
  - If R1, R2, …, Rn are a set of relation schemas, then we can write the relational database schema, or simply relational schema, R, as:

    $R = \{R_1, R_2, \ldots, R_n\}$
- Schema is the blueprint or structure that defines something

Properties of Relations

- 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.

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.
- **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).

Relational Keys

- **Primary Key**
  - Candidate key selected to identify tuples uniquely within relation.
- **Alternate Keys**
  - Candidate keys that are not selected to be primary key.
- **Foreign Key**
  - Attribute, or set of attributes, within one relation that matches candidate key of another relation. Used for cross-reference between tables.

Representing Relational Database Schemas

- A relational database consists of any number of normalized relations.
- The common convention for representing a relation schema is to give the name of the relation followed by the attribute names in parentheses. Normally, the primary key is underlined.
- The conceptual model, or conceptual schema, is the set of all such schemas for the database.

Representing Relational Database Schemas

The relational schema for part of the DreamHome case study:

Branch	(branchNo, street, city, postcode)
Staff	(staffNo, fName, lName, position, sex, DOB, salary, branchNo)
PropertyForRent	(propertyNo, street, city, postcode, type, rooms, rent, ownerNo, staffNo, branchNo)
Client	(clientNo, fName, lName, telNo, prefType, maxRent, eMail)
PrivateOwner	(ownerNo, fName, lName, address, telNo, eMail, password)
Viewing	(clientNo, propertyNo, viewDate, comment)
Registration	(clientNo, branchNo, staffNo, dateJoined)

Representing Relational Database Schemas

Instance of the DreamHome rental database (Staff relation):

Staff

staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

Integrity Constraints

- Integrity constraints ensure that the data is accurate.
- Domain constraints form restrictions on the set of values allowed for the attributes of relations.
- Two principal integrity rules for the relational model are entity integrity and referential integrity.

Integrity Constraints

- Before defining entity and referential integrity, let's introduce the concept of nulls.
- 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.

Integrity Constraints

- **Entity Integrity**
  - In a base relation, no attribute of a primary key can be null.
  - Relations that store data are called "base relations" and are implemented as "tables".
- **Referential Integrity**
  - If foreign key exists in a relation, either foreign key value must match a candidate key value of some tuple in its home relation or foreign key value must be wholly null.

Integrity Constraints

- General Constraints
  - Additional rules specified by users or database administrators that define or constrain some aspect of the enterprise.
  - For example: the number of staff that may work at a branch office (refer to the Staff relation) must not exceed 20

Views

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

- **View**
  - A virtual or derived relation
  - 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.

Purpose of Views

- Provides powerful and flexible security mechanism by hiding parts of database (attributes and tuples) 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.

Updating Views

- There are restrictions on types of modifications that can be made through views:
  - Updates are allowed if query involves a single base relation and contains a candidate key of base relation.
  - Updates are not allowed involving multiple base relations.
  - Updates are not allowed involving aggregation or grouping operations.

Updating Views

- Classes of views are defined as:
  - theoretically not updateable;
  - theoretically updateable;
  - partially updateable.