THE RELATIONAL MODEL

LECTURE 3

RELATIONAL DATA MODEL

- In the relational model, all data is logically structured within relations (tables).
- Each **relation** has a name and is made up of named **attributes** (columns) of data.
- Each **tuple** (row) contains one value per attribute.

THE RELATIONAL MODEL'S OBJECTIVES (CODD, 1970).

- To allow a high degree of data independence.
- To provide substantial grounds for dealing with data semantics, consistency, and redundancy problems.
- To enable the expansion of set-oriented data manipulation languages.

RELATIONAL DATA STRUCTURE TERMINOLOGY

- **Relation**
  - A relation is a table with columns and rows.
- **Attribute**
  - An attribute is a named column of a relation.
- **Domain**
  - A domain is the set of allowable values for one or more attributes.

EXAMPLE 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

RELATIONAL DATA STRUCTURE TERMINOLOGY

- **Tuple**
  - A tuple is a row of a relation.
- The structure of a relation, together with a specification of the domains and any other restrictions on possible values, is sometimes called its **intension**, which is usually fixed unless the meaning of a relation is changed to include additional attributes.
- The tuples are called the **extension** (or **state**) of a relation, which changes over time.

RELATIONAL DATA STRUCTURE TERMINOLOGY

- **Degree**
  - The degree of a relation is the number of attributes it contains.
    - Unary
    - Binary
    - Ternary
    - N-ary

RELATIONAL DATA STRUCTURE TERMINOLOGY

- **Cardinality**
  - The cardinality of a relation is the number of tuples it contains.
- **Relational database**
  - A collection of normalized relations with distinct relation names.

ALTERNATIVE TERMINOLOGY

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

DATABASE RELATIONS

- Relation schema
  - A named relation defined by a set of attributes and domain name pairs.
- Let $A1, A2, \ldots, A_N$ be attributes with domains $D1, D2, \ldots, D_N$. Then the set $\{A1{:}D1, A2{:}D2, \ldots, A_N{:}D_N\}$ is a relation schema.
- A relation R defined by a relation schema S is a set of mappings from the attribute names to their corresponding domains.
- Thus, relation R is a set of n-tuples: $(A1{:}d1, A2{:}d2, \ldots, AN{:}dn)$ such that $d1 \in D1$, $d2 \in D2, \ldots, dn \in D_N$

DATABASE RELATIONS

- RELATIONAL DATABASE SCHEMA
  - A SET OF RELATION SCHEMAS, EACH WITH A DISTINCT NAME.
- If *R*1, *R*2, . . . , *Rn* are a set of relation schemas, then we can write the *relational database schema*, or simply *relational schema*, *R*, as:

$$R = \{R1, R2, \ldots, RN\}$$

PROPERTIES OF RELATIONS

- The relation has a name that is distinct from all other relation names in the relational schema;
- Each cell of the relation contains exactly one atomic (single) value;
- Each attribute has a distinct name;
- The values of an attribute are all from the same domain;
- Each tuple is distinct; there are no duplicate tuples;
- The order of attributes has no significance;
- The order of tuples has no significance, theoretically. (However, in practice, the order may affect the efficiency of accessing tuples.)

RELATIONAL KEYS

- There is need to be able to identify one or more attributes (called **relational keys**) that uniquely identifies each tuple in a relation.
- **Superkey -** an attribute, or set of attributes, that uniquely identifies a tuple within a relation.

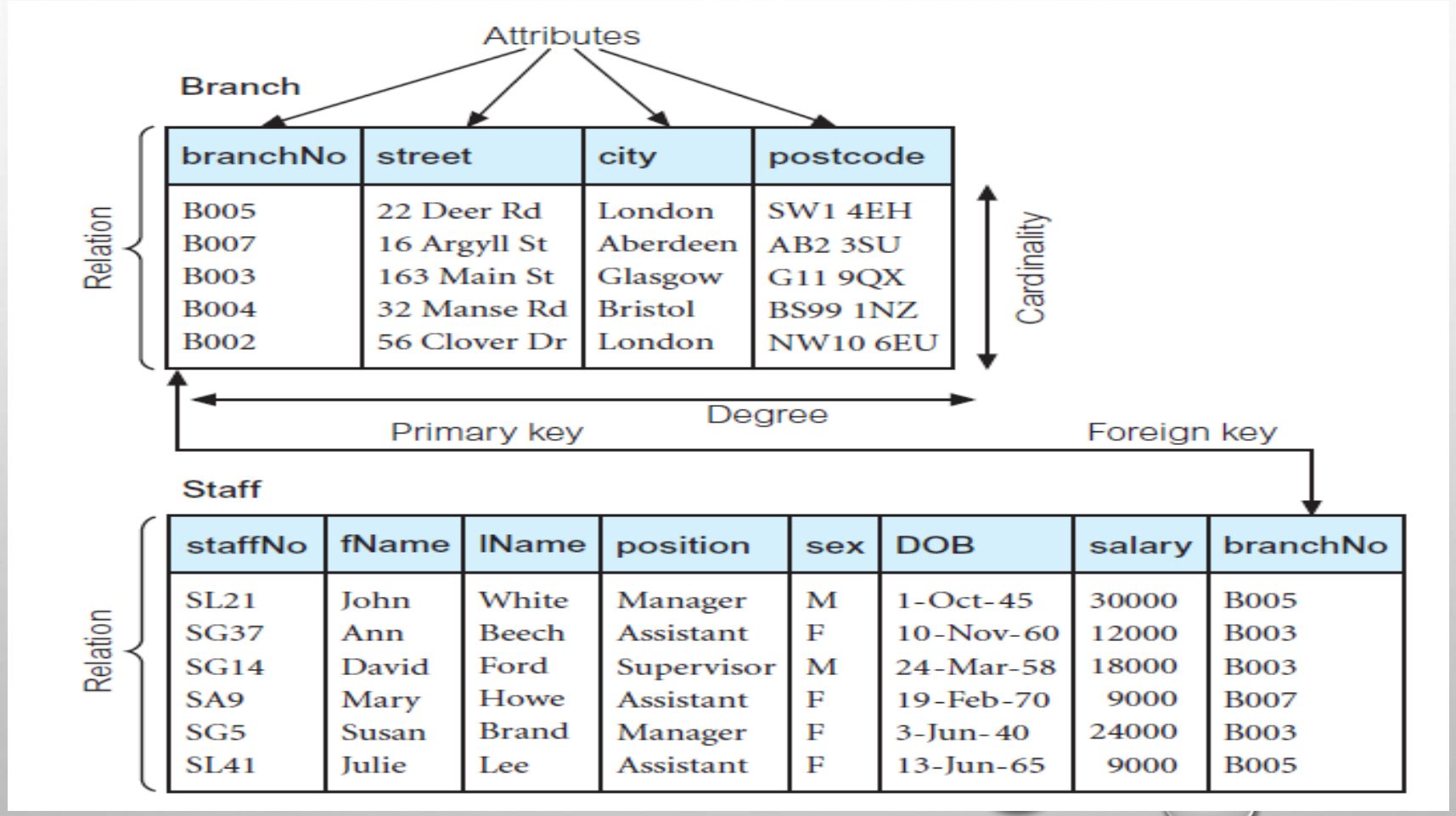
RELATIONAL KEYS

- **Candidate key**
- A superkey such that no proper subset is a superkey within the relation.
- Candidate key, *k*, for a relation *r* has two properties:
  - **Uniqueness** – in each tuple of *R*, the values of *K* uniquely identify that tuple;
  - **Irreducibility** – no proper subset of *K* has the uniqueness property.

RELATIONAL KEYS

- **PRIMARY KEY**
- THE CANDIDATE KEY THAT IS SELECTED TO IDENTIFY TUPLES UNIQUELY WITHIN THE RELATION.
- **ALTERNATE KEYS**
- THE CANDIDATE KEYS THAT ARE NOT SELECTED TO BE THE PRIMARY KEY
- **FOREIGN KEY**
- AN ATTRIBUTE, OR SET OF ATTRIBUTES, WITHIN ONE RELATION THAT MATCHES THE CANDIDATE KEY OF SOME RELATION.

BRANCH AND STAFF RELATIONS



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

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

EXAMPLE

- 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)
- Privateowner (ownerno, fname, lname, address, telno)
- Viewing (clientno, propertyno, viewdate, comment)
- Registration (clientno, branchno, staffno, datejoined)

RELATIONAL INTEGRITY CONSTRAINTS

- To ensure that data is accurate
- Domain constraints
- Integrity rules
  - Entity integrity
  - Referential integrity
- Multiplicity
- General constraints

NULLS

- Nulls represent a value for an attribute that is currently unknown or is not applicable for this tuple.
- Nulls are a way to deal with incomplete or exceptional data.
- Zeros and spaces are values, but a null represents the absence of a value.

INTEGRITY RULES

- **Entity integrity**
- In a base relation, no attribute of a primary key can be null.
- **Referential integrity**
- If a foreign key exists in a relation, either the foreign key value must match a candidate key value of some tuple in its home relation or the foreign key value must be wholly null.

PROBLEM IDENTIFICATION