TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

Database
Lesson 2. Relational databases

Ba Lam DO

Communication channel & course materials

https://www.facebook.com/groups/lamdb.soict

Nhóm sinh viên thầy Lâm

Learning Map

Sequence	Title
1	Introduction to Databases
2	Relational Databases
3	Relational Algebra
4	Structured Query Language – Part 1
5	Structured Query Language – Part 2
6	Constraints and Triggers
7	Entity Relationship Model
8	Functional Dependency
9	Normalization
10	Storage - Indexing
11	Query Processing
12	Transaction Management – Part 1
13	Transaction Management – Part 2

Outline

1. Relational data model
2. Constraints

Learning objectives

- Upon completion of this lesson, students will be able to:
  - Recall some basic concepts of relational data model.
  - Show some constraints of relational data model.

Keywords

Data model	A set of concepts used to describe the structure of a database: data types, relationships, constraints, semantics...
Relation	Is thought of as a table of values, each row in the table represents a collection of related data values.
Key	An attribute or a set of attributes in the relation, which can identify a tuple uniquely.
Integrity constraints	Provide a way of ensuring that changes made to the database by aut horized users do not result in a loss of data consistency.

1. Relational data model

1.1. Introduction

1.2. Database Basic concepts

1.1. Different data models

- Hierarchical database model
- Network model
- Object-oriented database model
- Relational model
- Entity-relationship model
- Document model
- …

1.2. Relational data model

- Is very simple model, was first introduced by Ted Codd of IBM Research in 1970.
- Used by most of commercial database systems.
- Query with high-level languages.
- Efficient implementations.
- Based on mathematical theory, closed to file structure and data structure, there are three sets of terminology:

Relation	Table	File
Tuple	Row	Record
Attribute	Column	Field

1.3. Basic concepts

Relations	• are saved in the format of tables, which have rows and columns.
Relation instance/state	• actual contents at given point in time. The lowercase letter s q, r, s denote relation states.
Database	• a set of named relations (or tables).

1.3. Basic concepts [2]

Tuple	• a single row of a table, which contains a single record for t hat relation. The letters t, u, v denote tuples.
Cardinality	• is the number of tuples in a relation.
Degree (arity)	• is the number of attributes in a relation.

1.4. Relational schema

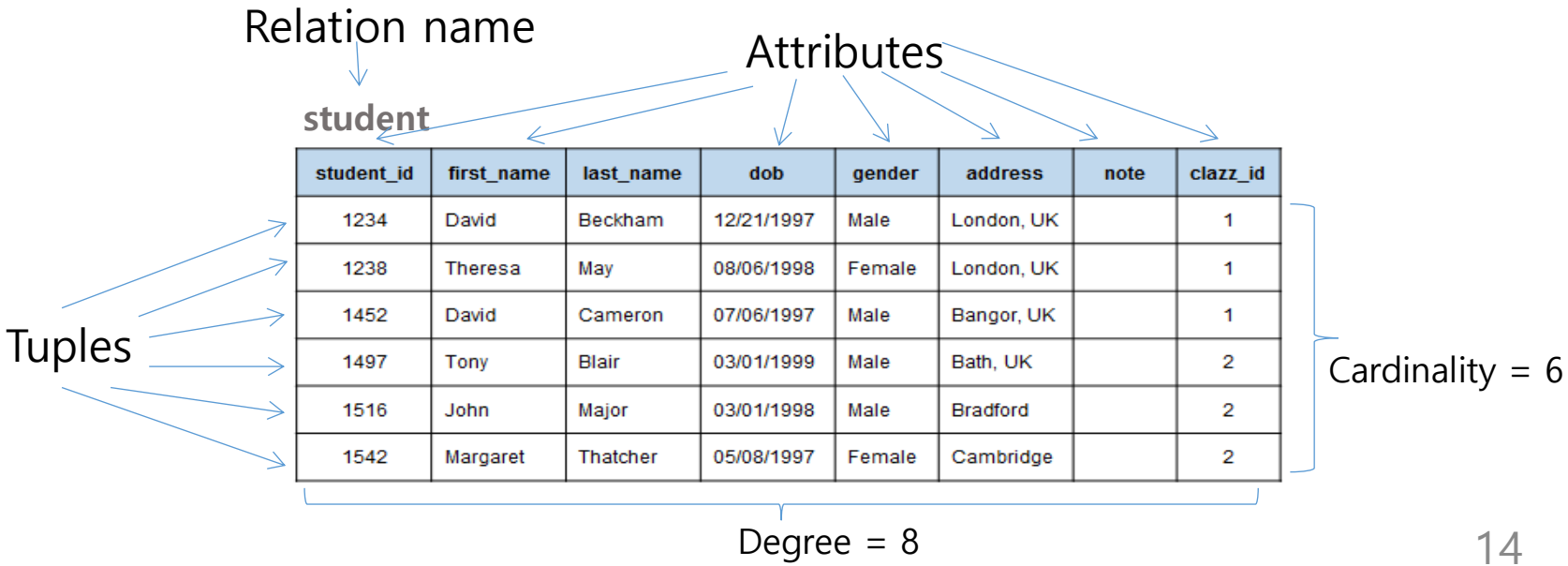
- Relational schema: structural description of relations in database.
- A relation schema R of degree n, denoted by $R(A_1, A_2, ..., A_n)$, is made up of a relation name R and a list of attributes $A_1, A_2, ..., A_n$
- Each attribute A_i has values belong to domain Di of Ai, denoted by $dom(A_i)$
- An n-tuple t in a relation r(R) is denoted by $t = <v_1, v_2, ..., v_n>$, where v_i is the value corresponding to attribute A_i. Both $t[A_i]$ and $t.A_i$ (and sometimes t[i]) refer to the value v_i in t for attribute A_i

1.4. Relational schema [2]

- Notice that the uppercase letters Q, R, S denote relation names.
- A relation (or relation state) r of the relation schema $R(A_1, A_2, ..., A_n)$, also denoted by r(R), is a set of n-tuples $r = \{t_1, t_2, ..., t_m\}$. Each n-tuple t is an ordered list of n-values $t = <v_1, v_2, ..., v_n>$, where each value v_i, $1 \leq i \leq n$, is an element of $dom(A_i)$ or is a special NULL value.
- A relation (or relation state) r(R) is a mathematical relation of degree n on the domains $dom(A_1), dom(A_2), ..., dom(A_n)$, which is a subset of the Cartesian product of the domains that define R: $r(R) \subseteq (dom(A_1) \times dom(A_2) \times ... \times dom(A_n))$

1.5. An example

student(student_id, first_name, last_name, dob, gender, address, note, clazz_id)



student_id	first_name	last_name	dob	gender	address	note	clazz_id
1234	David	Beckham	12/21/1997	Male	London, UK		1
1238	Theresa	May	08/06/1998	Female	London, UK		1
1452	David	Cameron	07/06/1997	Male	Bangor, UK		1
1497	Tony	Blair	03/01/1999	Male	Bath, UK		2
1516	John	Major	03/01/1998	Male	Bradford		2
1542	Margaret	Thatcher	05/08/1997	Female	Cambridge		2

2. Constraints

2.1. Introduction

2.2. Types of constraints

2.3. An example

2.1. Introduction

- Every relation has some conditions that must hold for it to be a valid relation.
- These conditions are called **Relational Integrity Constraints**.
- Provide a way of ensuring that changes made to the database by authorized users do not result in a loss of data consistency.

2.2. Types of constraints

- Key constraints
- Domain constraints
- Referential integrity constraints

2.2.1. Key constraints

- A key is an attribute or a set of attributes in the relation, which can identify a tuple uniquely.
- Key constraints enforce the following
  - in a relation with a key, no two tuples can have identical values for key attributes.
  - a key can not have NULL values.
  - Key constraints are also referred to as Entity Constraints.

2.2.1. Key constraints [2]

- Superkey: An attribute, or a 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 the relation, values of K uniquely identify that tuple (uniqueness)
  - No proper subset of K has the uniqueness property (irreducibility)
- Primary Key: Candidate key selected to identify tuples uniquely within a relation. Each key attribute of primary key has its name underlined.

2.2.1. Key constraints [3]

- Alternate Keys: Candidate keys that are not selected to be the primary key.
- Minimal key: a minimal set of attributes that can be used to identify a single tuple.
- Foreign Key:
  - Attribute, or set of attributes, within one relation that matches candidate key of some relation
  - Used to model relationships between relations
  - Each key attribute of foreign key has its name italic

2.2.2. Domain constraints

- Attributes have specific values in real-world scenario. Every attribute is bound to have a specific range of values.
- Within each tuple, the value of each attribute A must be an atomic value from the domain dom(A).
- The data types associated with domains:
  - standard numeric data types for integers (short integer, integer, and long integer) and real numbers (float, double precision float).
  - Characters, Booleans, fixed-length strings, and variable-length strings, date, time, timestamp, and money, or other special data types.
  - a subrange of values from a data type .
  - an enumerated data type in which all possible values are explicitly listed.

2.2.2 Domain constraints [2]

- Null value
  - Represents value for an attribute that is currently unknown or not applicable for any tuple;
  - deals with incomplete or exceptional data;
  - represents the absence of a value and is not the same as zero or spaces

2.2.2. Referential integrity constraints [3]

- Referential integrity constraints
  - Referential integrity constraints work on the concept of Foreign Keys. A foreign key is a key attribute of a relation that can be referred in other relation.
  - Referential integrity constraint states that if a relation refers to a key attribute of a different or same relation, then that **key element must exist.**

2.3. An example

- student(**student_id**, first_name, last_name, dob, gender, address, note, *clazz_id*)
- subject(**subject_id**, name, credit, percentage_final_exam)
- enrollment(***student_id, subject_id, semester,*** midterm_score, final_score)

2.3. An example [2]

student

student_id	first_name	last_name	dob	gender	address	note	clazz_id
1234	David	Beckham	12/21/1997	Male	London, UK		1
1238	Theresa	May	08/06/1998	Female	London, UK		1
1452	David	Cameron	07/06/1997	Male	Bangor, UK		1
1497	Tony	Blair	03/01/1999	Male	Bath, UK		2
1516	John	Major	03/01/1998	Male	Bradford		2
1542	Margaret	Thatcher	05/08/1997	Female	Cambridge		2

subject

subject_id	name	credit	percentage_final_exam
IT3090	Databases	3	0.7
IT4843	Data integration	3	0.7
IT4868	Web mining	2	0.6
IT2000	Introduction to ICT	2	0.5
IT3020	Discrete Mathematics	2	0.7
IT3030	Computer Architectures	3	0.7

Foreign key

Foreign key

enrollment

student_id	subject_id	semester	midterm_score	final_score
1234	IT3090	20171	7	8
1238	IT3090	20171	9	8
1452	IT3090	20171	6	6
1234	IT2000	20162	5	8
1234	IT3020	20171	8	9
1452	IT3030	20171	7	9
1238	IT3020	20162	7	7

Primary key

Remark

- Relational data model
- Constraints
  - Key constraints
  - Domain constraints
  - Referential integrity constraints

Summary

- Relational data model
  - Relations, relation instance/state, relation schema
  - Database, tuple
  - Cardinality, degree
- Constraints
  - Key constraints
  - Domain constraints
  - Referential integrity constraints

Next lesson: Relational Algebra

- Raghu Ramakrishnan and Johannes Gehrke, Database Management S ystems, 3rd edition, Mc Graw Hill, 2003.
- Elmasri and Navathe, Fundamentals of Database Systems, 6th edition , Addison-Wesley, 2011.