

# *Relational Database Model*

© Department of Computer Science  
Northern Illinois University  
September 2018

# *Basic Structure*

- Relation or Table
  - Named
  - NO repeating fields
  - Shown as Relation-Name (A1, A2, ....., An)

Relation Name

| A1 | A2 | .. | .. | .. | .. | .. | An |
|----|----|----|----|----|----|----|----|
|    |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |

# *Basic Structure*

- Columns
  - named attributes
  - must be atomic values
  - values valid within a domain

Relation Name

| A1 | A2 | .. | .. | .. | .. | .. | An |
|----|----|----|----|----|----|----|----|
|    |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |

# *Basic Structure*

- Rows
  - also called tuples
  - similar to record
  - must have primary key

Relation Name

| A1 | A2 | .. | .. | .. | .. | .. | An |
|----|----|----|----|----|----|----|----|
|    |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |

# *Basic Structure*

- Keys
  - Super Key:
    - an attribute or set of attributes that uniquely identify a tuple
    - every relation has at least one superkey, the set of all attributes
    - a relation can have more than one superkey

# *Basic Structure*

- Keys
  - Candidate Key:
    - a minimum set of attributes that uniquely identify a tuple
    - a minimal super key
    - a relation may have more than one candidate key
  - Primary Key:
    - one and only one per relation.
    - a chosen candidate key

# *Basic Structure*

- Keys Example
  - Employee ( Emp-ID, Emp-Name, Emp-Birthdate, Emp-Address, Emp-Salary)
    - Super key:
      - Emp-ID, Emp-Address
      - Emp-Name, Emp-Birthdate, Emp-Salary
    - Candidate key:
      - Emp-ID
      - Emp-Name, Emp-Birthdate
    - Primary key:
      - Emp-ID

# *Basic Structure*

- Keys Example
  - Employee-Project( Emp-ID, Project-ID, Emp-Title-Proj, Hours-Worked )
    - Super key:
      - Emp-ID, Project-ID, Emp-Title-Proj
      - Emp-ID, Project-ID, Hours-Worked
    - Candidate key:
      - Emp-ID, Project-ID
      - Project-ID, Emp-Title-Proj (assuming each employee has a different title within a project)
    - Primary key:
      - Emp-ID, Project-ID



# Basic Structure

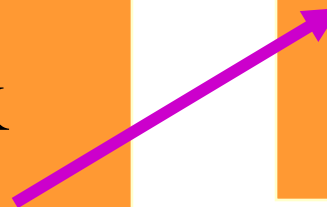
- Foreign Key
  - used to reference another relation
  - attributes of FK have same domain as the primary key of the home relation
  - example

Section

| <u>Sect-ID</u> | Sect-Time | Crse-ID |
|----------------|-----------|---------|
| PK             |           | FK      |

Course

| <u>Crse-ID</u> | Crse-Title | .... |
|----------------|------------|------|
| PK             |            |      |



# Basic Structure

- Foreign Key

Course

| <u>Crse-ID</u> | Crse-Title |  |
|----------------|------------|--|
| PK             |            |  |

Student

| <u>Student-SSN</u> |  |
|--------------------|--|
| PK                 |  |

completed

| <u>Crse-ID</u> | <u>Student-SSN</u> |  |
|----------------|--------------------|--|
| PK             | PK                 |  |
| FK             | FK                 |  |

# *Terminology*

- Domain
  - set of atomic valid values of one or more attribute
  - may be specified as a data type
- Atomic values
  - indivisible data values
- Null value
  - designates a MISSING attribute value
  - may or may not be allowed for an attribute

# *Null Values*

- Does the value exist?
- Is the value known?
- Data – exists and known (Not a NULL value)
- NULL values
  - Missing – exists, but not known
  - N/A – doesn't exist, but known not to exist
  - Unknown – existence? value?

# *Terminology*

- Degree
  - number of attributes (columns) in a relation
  - does not changes dynamically
- Cardinality
  - number of tuples (rows) in a relation
  - changes dynamically with additions and deletions of tuples using DML

# *Terminology*

- Intention
  - a named relation and its attribute names
  - also called schema of a relation
  - the DDL is used to modify the intention
- Extension
  - the data (tuples) in a relation
  - the state of a relation
  - the DML is used to modify the extension

# *Characteristics of a Relation*

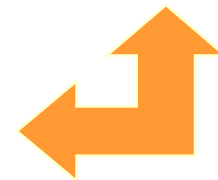
- Order Independence
- Two kinds
  - (1) ordering of tuples within a relation
    - do not have any particular order
    - considered an unordered set

# Characteristics of a Relation

- Order Independence
  - (2) ordering of attributes within a relation
    - do not have any particular order as long as correspondence between the attribute and its values is maintained
    - Example

Student(Stud-ID, Stud-Name, Stud-Address)

Student(Stud-Address, Stud-Name, Stud-ID)



equivalent  
relations



# *Relational Constraints*

- Domain / Integrity Constraints
  - specify the valid values of each attribute
  - editing criteria
    - salary not  $> 100k$
    - height  $< 8$  feet

# *Relational Constraints*

- Entity Integrity Constraint
  - states that no attribute of a primary key can contain a null value
    - Game ( Date, Location, Time)
      - here neither Date nor Location nor both can ever contain a null value in this relation

# Relational Constraints

- Referential Integrity Constraint
  - a foreign key can
    - EITHER contain an existing value of the PK in the home relation
    - OR contain a NULL value

| <u>Sect-ID</u> | Sect-Time | C |
|----------------|-----------|---|
| S1             | 10:00am   | C |
| S2             | 2:00pm    | C |
| S3             | 3:00pm    | C |

| <u>Crse-ID</u> | Crse-Title | ... |
|----------------|------------|-----|
|                |            | ..  |
| C1             | Course 1   |     |
| C2             | Course 2   |     |
| C2             | Course 2   |     |

# Relational Constraints

Crse-ID in Section relation is the Foreign Key

- Referential Integrity Constraints

- a foreign key can

- EITHER contain a valid value of the PK in the home relation
- OR contain a NULL value

| <u>Sect-ID</u> | Sect-Time | Crse-ID |
|----------------|-----------|---------|
| S1             | 10:00am   | C1      |
| S2             | 2:00pm    | C1      |
| S3             | 3:00pm    | C2      |

Crse-ID in Course relation is the referenced Primary Key

| <u>Crse-ID</u> | Crse-Title | ... |
|----------------|------------|-----|
| C1             | Course 1   | ..  |
| C2             | Course 2   |     |
| C3             | Course 3   |     |