Relational Database Basics Review
- Overview
- Database approach
- Database system
- Relational model

File Processing Approaches
- Based on file systems
  - Data are recorded in various types of files organized in folders (directories)
- File types
  - Sequential data files
  - Name-value pair files
  - Spreadsheets or list files
  - XML files

Files Processing Problems
- Difficult to handle complex data
- Low data quality: redundancy and inconsistency
- No central management
- Difficult to maintain and share in multi-user environments

Database Approach
- Database is a structured and self-describing collection of data
- Structured: structures and rules are consistently and rigorously defined and enforced (integrity)
- Self-describing: the description of data (data definition, or metadata) is contained within the database

Database Approach
- Centralized management
- Managed and controlled by specialized programs, called database management systems (DBMS),
  - which provides rich data management functionalities
Advantages of Databases
- High data quality, integrity, and consistency
- Reduced data redundancy and application maintenance
- Easy access and sharing
- Scalable
- Improved security
- Specialized and productive management tool

Major disadvantages
- Increased complexity
- Greater impact of failure

Database System
- A database system is a complete information system
- Basic layers of a database system

Database
- A database is a storage place for data
- What's in the database?
  - Data (tables)
  - Metadata
  - Other data and structures

Metadata
- Metadata are data that describe data (data definitions)
- Metadata is always a part of a database.

Database Management System
- DBMS serves as a controller (gatekeeper) for databases
- DBMS provides common functionalities and interfaces for managing and controlling database activities, such as
  - creating and maintaining databases and other structures
Database Management System
- reading, updating and deleting data
- data backup and recovery
- controlling concurrency, consistency, and enforcing other rules
- providing security

Introduction to Relational Databases
- The relational approach was originally proposed in 1970’s.
- The first project that proved the practicality of the relational model is System R, developed at IBM’s San Jose Research Laboratory in 1976. Examples: Oracle, Sybase, access

Introduction to Relational Databases cont:
- The relational model will have only values.
- Even references between data in different sets (relations) are represented by means of values.
- In the hierarchical and network model there are explicit references (pointers), which make them more complicated.

Enterprise Database System

Introduction to Relational Databases cont:
- The model was proposed as a disciplined way of handling data using the rigour of mathematics, particularly set theory.
- This would enhance the concept of program-data independence
- and improve programmer activities.

Relational Data Structure
- The Relational approach is based on elementary mathematical relation theory.
- Its basic construct is a relation. A relation is also called a table.
- The data is organized in tables. The table has columns and rows.
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 Cont:

- 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.

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
- A key is one or more columns of a relation that is used to uniquely identify a record
  - Primary key
  - Candidate key
  - Alternate key
  - Surrogate key
  - Composite key
  - Foreign key

Candidate Key
- The minimum set of column(s) that uniquely identifies a single record (row)
- Each value in this column is unique in this relation

Primary Key
- Primary key is a column/attribute that is used to uniquely identify a record
- Is one of the candidate keys chosen to be the identifying key; others become alternate keys
- Each value of this key column uniquely identifies a single record (row)
- There is only ONE primary key for a table

Composite Key
- A composite key contains two or more attributes (columns)
- Example:
  - “FirstName” + “LastName”
  - “FirstName” + “LastName” + “BirthDate”
  - “FirstName” + “LastName” + “BirthDate” + “BirthCity”

Artificial Primary Key/Surrogate Key
- Sometimes it is difficult to find a natural attribute as a primary key, or it is difficult to use a composite key.
- A column is created arbitrarily and assign each record a unique number/id
- Product Number, Product Id, Movie Id, Actor Id, etc.
**Artificial Primary Key/Surrogate Key**

<table>
<thead>
<tr>
<th>AdviserID</th>
<th>AdviserName</th>
<th>Department</th>
<th>Phone</th>
<th>Office</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Johnson</td>
<td>Biology</td>
<td>236-9079</td>
<td>Sci123</td>
</tr>
<tr>
<td>2</td>
<td>Wu</td>
<td>Chemistry</td>
<td>236-0091</td>
<td>Sci260</td>
</tr>
<tr>
<td>3</td>
<td>Horan</td>
<td>Math</td>
<td>236-0088</td>
<td>AR-45</td>
</tr>
</tbody>
</table>

**PK Selection Guidelines**

- Do not use a field whose value is frequently changed as PK
- Look for single-attribute PK first
- If a PK contains more than 3 columns, consider a surrogate key
- Don't be limited to sample data; think beyond and consider possible scenarios and requirements

**Relationship and Foreign Key (FK)**

- Relationship is how tables (relations) are linked
- It is defined by the foreign key (FK) constraint
- A foreign key references a primary key (or any other unique keys) in another table
- This pair of keys are of the same kind (may be of different name)

**Relationship in SQL Server Database**

Relationship and FK Example

```sql
CREATE TABLE Retail(
    ItemID INT PRIMARY KEY,
    ItemName VARCHAR(50)
)
CREATE TABLE Order(
    OrderID INT PRIMARY KEY,
    ItemID INT
)
```