



Basic Terminologies and ER Model

Basic Terminologies in DBMS

- **Data:** Data is any sort of information which is stored in computer memory
- **Database:** A collection of information (tables) related to a particular topic or purpose
 - There are two types of databases: Nonrelational and relational
 - A **nonrelational database** is a database that does not use the tabular schema of rows and columns found in most traditional database systems
 - Instead, non-relational databases use a storage model that is optimized for the specific requirements of the type of data being stored
 - Called NoSQL databases
 - Also called a flat file, stores information in one table
 - Nonrelational databases are useful for information stored in a single list, such as a list of student names, addresses, and phone numbers
 - The most popular being MongoDB, DocumentDB, Cassandra, Couchbase, HBase, Redis, and Neo4j

Basic Terminologies in DBMS

- A **relational database** is a collection of data items with pre-defined relationships between them
 - These items are organized as a set of tables with columns and rows
 - Tables are used to hold information about the objects to be represented in the database
 - Also be called relational database management systems (RDBMS) or SQL databases
 - The most popular of these are Microsoft SQL Server, Oracle Database, MySQL, and IBM DB2
- **Database management system:** A Database Management System (DBMS) is software designed to store, retrieve, define, and manage data in a database
- **Database model:** The structure of the information stored in the database
 - This model should included how each individual piece of information relates to all the other information in the database
 - Proper planning, even in the initial pencil-and-paper stage, ensures that the database you create and maintain is efficient and provides easy access to the information you need most
 - A well-designed database should eliminate the need to enter the same data repeatedly and prevent duplication of information, thereby maintaining the integrity of the data

Basic Terminologies in DBMS

- **Database modeling:** The process of strategically planning where to store each piece of information you wish to include in your database
- **Information:** It is organized or classified data, which has some meaningful values for the receiver
 - Information is the processed data on which decisions and actions are based
 - **Data** is raw, unprocessed, unorganized **facts** that are seemingly random and do not yet carry any significance or meaning
 - **Information** refers to data that has been organized, interpreted, and contextualized by a human or machine so that it possess relevance and purpose
- **Record/Tuple/Row:** A **record** is a **database** entry that may contain one or more values
 - Groups of **records** are stored in a table, which defines what types of data each **record** may contain
 - Databases may contain multiple tables which may each contain multiple **records**

Basic Terminologies in DBMS

- **Field/Attribute/Column:** In a database table, a **field** is a data structure for a single piece of data
 - Fields are organized into records, which contain all the information within the table relevant to a specific entity
 - **Fields** appear as columns in a table and as cells in a form
 - For example, in a table called *customer contact information*, telephone number would likely be a field in a row that would also contain other fields such as street address and city
 - The records make up the table rows and the fields make up the columns
- **Datasheet view:** **Datasheet view** refers to row wise and column wise viewing of data in a table in database applications such as spreadsheets, Access, Excel, and so on
 - The information pertaining to individual records is provided in individual rows and the attributes related to that record is given in the corresponding columns

Basic Terminologies in DBMS

- **Field data types:** A characteristic of a field that determines what kind of data it can store
 - For example, a field whose data type is *Text* can store data consisting of either text or number characters, but a *Number* field can store only numerical data
- **Field list:** A small window that lists the fields of a selected table or data source
- **Recordset:** The set of records and fields that result from running a query
- **Form:** A structured document with specific areas for viewing or entering data one record at a time
 - Forms can be constructed in columnar, tabular, datasheet, or a simple justified format
 - Displays data from a table or a query one record at a time
- **Join line:** The line between two tables identifying the common field between them
- **Object:** A component of a database, such as a table, query, form, or report

Basic Terminologies in DBMS

- **Relationship:** A relationship is a situation that exists between two relational database tables when one table has a foreign key that references the primary key of the other table
 - The direct or indirect association between any two tables in a database
 - Relationships allow relational databases to split and store data in different tables, while linking disparate data items
- **One-to-many relationship:** A relationship in which a record in the primary table can be related to one or more records in the related table
- **One-to-one relationship:** A relationship between two tables in which for each record in the first table, there is only one corresponding record in the related table
- **Related table:** A table with a common field that uses values stored in a primary table
- **Primary key:** A field in a table whose value is uniquely identifies each record in the table

Basic Terminologies in DBMS

Cardinality: 12

Degree: 4

Columns

Field data type: Number

Rows

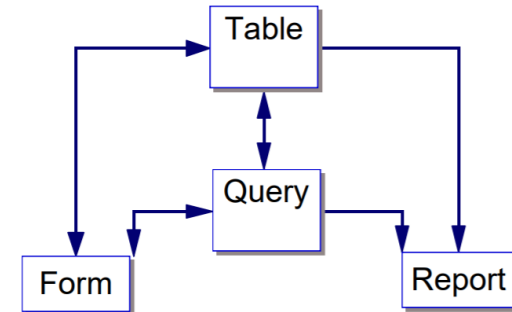
<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

The *instructor* table

Basic Terminologies in DBMS

- **Query:** A request for a particular collection of data in a database
 - Lists specific fields and records from a table based on selective criteria
- **Select Query:** A query that answers a question about one or more tables by limiting the number of records and fields displayed
- **Report:** A formatted collection of information organized to provide printed data on a specific subject
 - Presents data from a table or query in printable format
- **Page:** A page is a unit of storage whose size is configurable on a system-wide
 - Database pages are the internal basic structure to organize the data in the database files
- **Macro:** Automates a repetitive series of commands or group of commands as a single unit

Data Flow Diagram:



Overview of Database Design

- Conceptual design: (ER Model is used at this stage)
 - What are the entities and relationships in the enterprise?
 - What information about these entities and relationships should we store in the database?
 - What are the integrity constraints or business rules that hold?
 - A database 'schema' in the ER Model can be represented pictorially (ER diagrams)?
 - Can map an ER diagram into a relational schema?

ER Model Basics

- The ER data model was developed to facilitate database design by allowing specification of an enterprise schema that represents the overall logical structure of a database
- A database can be modeled as:
 - A collection of entities
 - Relationship among entities
- The ER data model employs three basic concepts:
 - Entity Sets
 - Relationship sets
 - Attributes
- The ER model also has an associated diagrammatic representation, the ER diagram, which can express the overall logical structure of a database graphically

Entity Sets

- An entity is an object that exists and is distinguishable from other objects
- Example: specific person, company, event, plant
- An entity set is a set of entities of the same type that share the same properties
- Example: set of all persons, companies, trees, holidays
- An entity is represented by a set of attributes; i.e., descriptive properties possessed by all members of an entity set
- Example:
 - instructor = (ID, name, salary)
 - student = (ID, name, dept)
- A subset of the attributes form a primary key of the entity set; i.e., uniquely identifying each member of the set

76766	Crick
45565	Katz
10101	Srinivasan
98345	Kim
76543	Singh
22222	Einstein

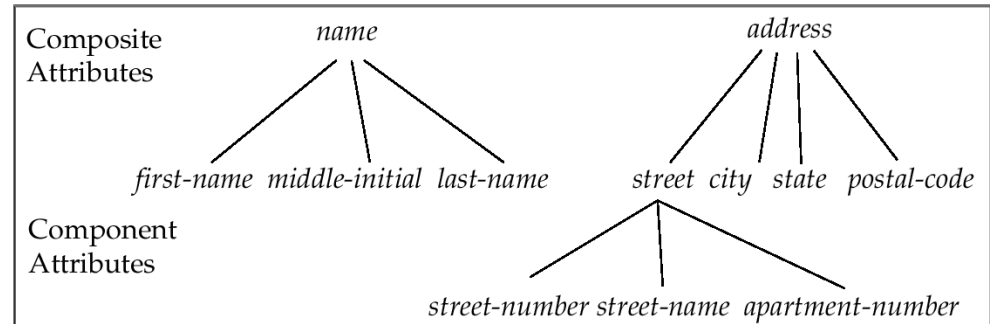
instructor

98988	Tanaka
12345	Shankar
00128	Zhang
76543	Brown
76653	Aoi
23121	Chavez
44553	Peltier

student

Attributes

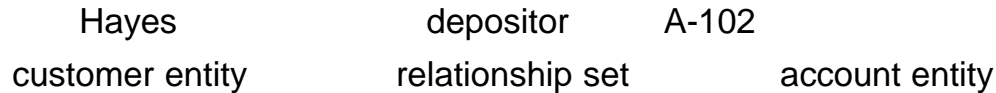
- An entity is represented by a set of attributes, that is descriptive properties possessed by all members of an entity set
- Domain: The set of permitted values for each attribute
- Attribute types:
 - Simple and composite attributes
 - Single-valued and multi-valued attributes
 - E.g., multivalued attribute: phone-numbers
 - Derived attributes
 - Can be computed from other attributes
 - E.g. age, given date of birth
- Composite attributes
- Component attributes



Relationship Sets

- A relationship is an association among several entities

- Example:



- A relationship set is a mathematical relation among $n \geq 2$ entities, each taken from entity sets

$$\{(e_1, e_2, \dots, e_n) \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n\}$$

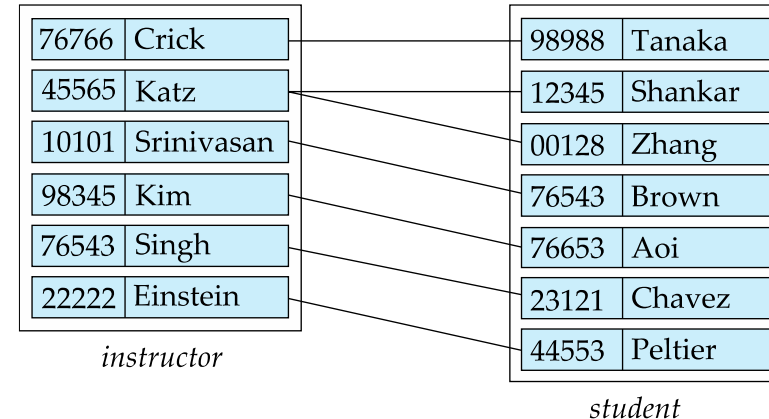
- where (e_1, e_2, \dots, e_n) is a relationship

- Example:

$(\text{Hayes}, \text{A-102}) \in \text{depositor}$

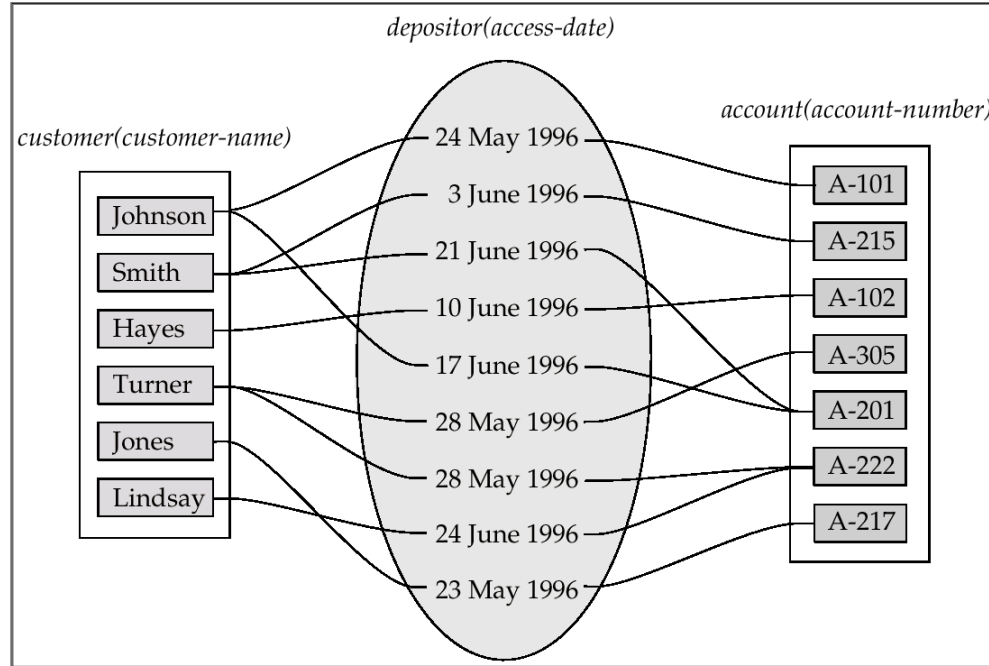
- Example: We define the relationship set **advisor** to denote the associations between **students** and the **instructors** who act as their advisors

- Pictorially, we draw a line between related entities



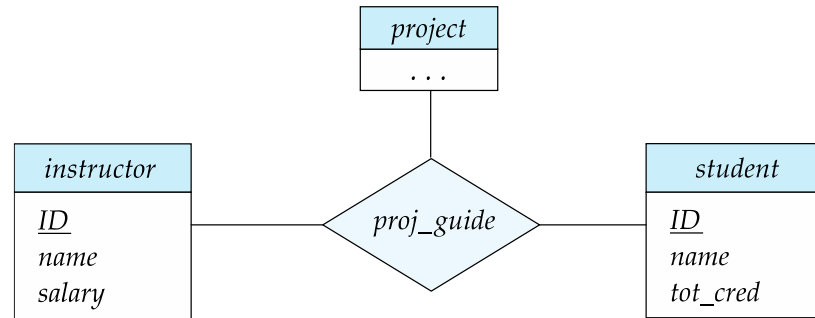
Relationship Sets

- An attribute can also be property of a relationship set
- For instance, the depositor relationship set between entity sets customer and account may have the attribute access-date



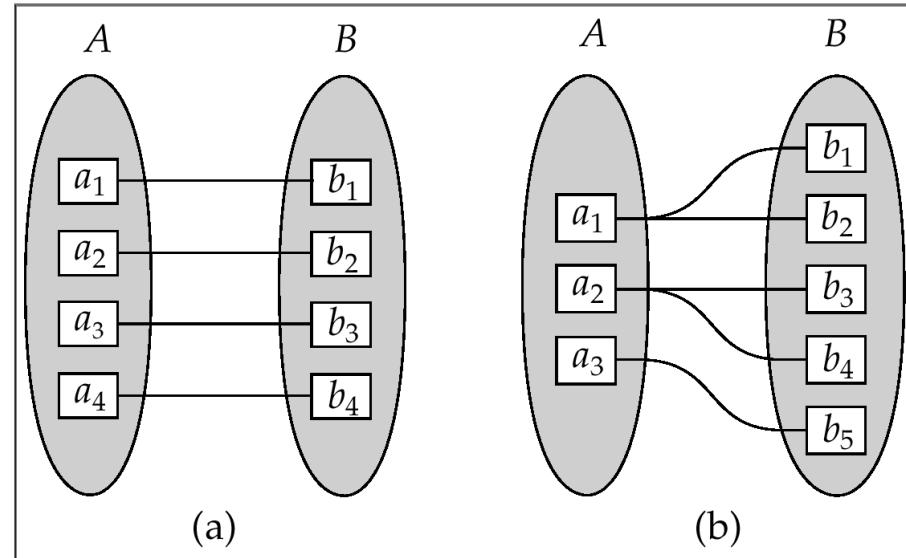
Degree of a Relationship Set

- Refers to number of entity sets that participate in a relationship set
- Relationship sets that involve two entity sets are binary (or degree two)
- Generally, most relationship sets in a database system are binary
- Relationship sets may involve more than two entity sets
 - E.g., Suppose employees of a bank may have jobs (responsibilities) at multiple branches, with different jobs at different branches
 - Then there is a ternary relationship set between entity sets *employee*, *job* and *branch*
- Relationships between more than two entity sets are rare



Mapping Cardinalities

- Express the number of entities to which another entity can be associated via a relationship set
- Most useful in describing binary relationship sets
- For a binary relationship set the mapping cardinality must be one of the following types:
 - One to one
 - One to many
 - Many to one
 - Many to many



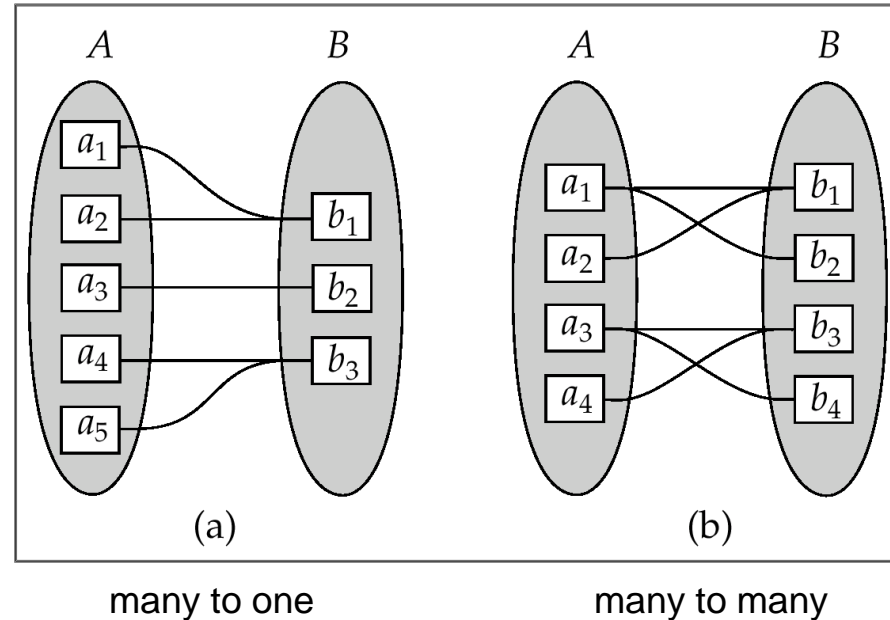
one to one

one to many

Note: Some elements in A and B may not be mapped to any elements in the other set

Mapping Cardinalities

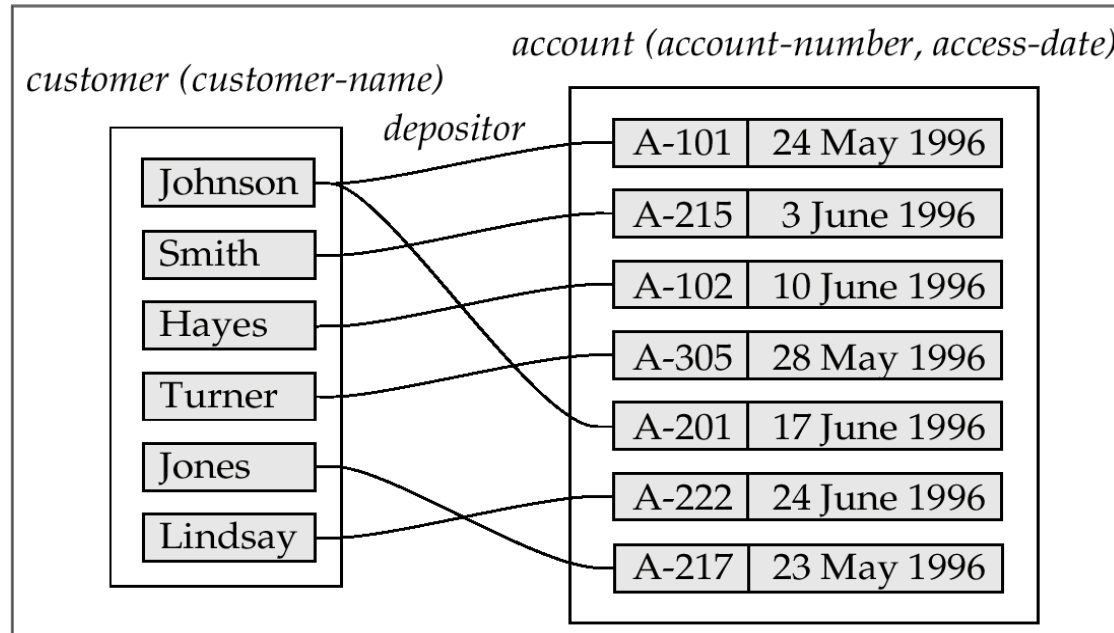
- Express the number of entities to which another entity can be associated via a relationship set
- Most useful in describing binary relationship sets
- For a binary relationship set the mapping cardinality must be one of the following types:
 - One to one
 - One to many
 - Many to one
 - Many to many



Note: Some elements in A and B may not be mapped to any elements in the other set

Mapping Cardinalities affect ER Design

- Can make access-date an attribute of account, instead of a relationship attribute, if each account can have only one customer
- The relationship from account to customer is many to one, or equivalently, customer to account is one to many



Next Lecture

ER Diagram

Thank you for your attention...

Any question?

Contact:

Department of Information Technology, NITK Surathkal, India
6th Floor, Room: 13

Phone: +91-9477678768

E-mail: shrutilipi@nitk.edu.in