ER DATA MODEL CONCEPTS

Muhammet Rasit Saygili
M.Vildan Sarikaya

Data Science
Clarusway

October 26, 2020

Outline

CLARUSWAY
WAY TO REINVENT YOURSELF

Database

Relational Database

A relational database is a type of database that stores and provides access to data points that are related to one another.

Figure 1: Relational Database

Relational Data Model

CLARUSWAY
WAY TO REINVENT YOURSELF

Relational Model (RM) or Entity Relationship (ER) Data Model represents the database as a collection of relations.

- Research on the theory of data/relationship/constraint
- Numerous database design methodologies
- The standard database access language called structured query language (SQL)
- Almost all modern commercial database management systems

Relational Data Model

CLARUSWAY
WAY TO REINVENT YOURSELF

ER modelling is based on two concepts:

- Entities, defined as tables that hold specific information (data)
- Relationships, defined as the associations or interactions between entities

Entity Relationship(ER) Diagram

CLARUSWAY
WAY TO REINVENT YOURSELF

ER Diagram is the visual representation of different entities within a system and how they relate to each other.

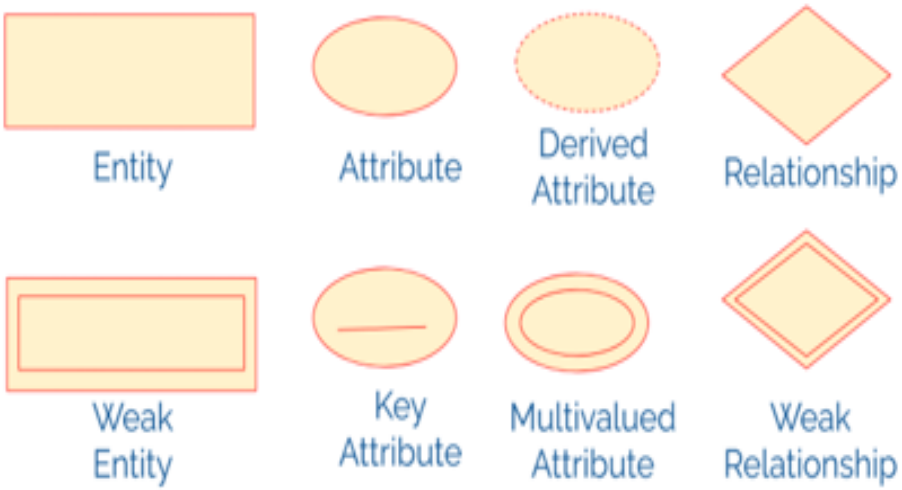
Figure 2: Entity Relationship(ER) Diagram

ERD Notations

CLARUSWAY
WAY TO REINVENT YOURSELF

There are several Notations for ER Diagrams. Most widely used ones are:

Chen's Notation:



Crow's Foot Notation:

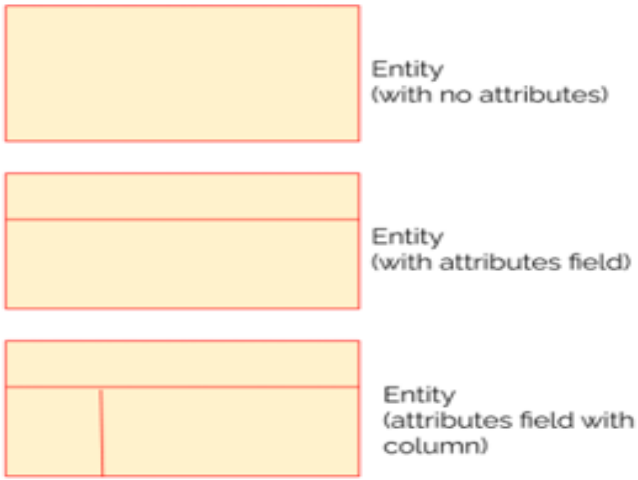


Figure 3: Chen's Notation and Crow's Foot Notation

Entity

An entity is an object in the real world with an independent existence that can be differentiated from other objects. An entity might be:

- An object with physical existence (e.g., a lecturer, a student, a car).
- An object with conceptual existence (e.g., a course, a job, a position).

Entity

CLARUSWAY
WAY TO REINVENT YOURSELF

Student

Roll_no	Student_name	Age	Mobile_no
1	Andrew	18	7089117222
2	Angel	19	8709054568
3	Priya	20	9864257315
4	Analisa	21	9847852156

Entity

Figure 4: Entity Table

Entity Type

The entity type is a collection of the entity having similar attributes.

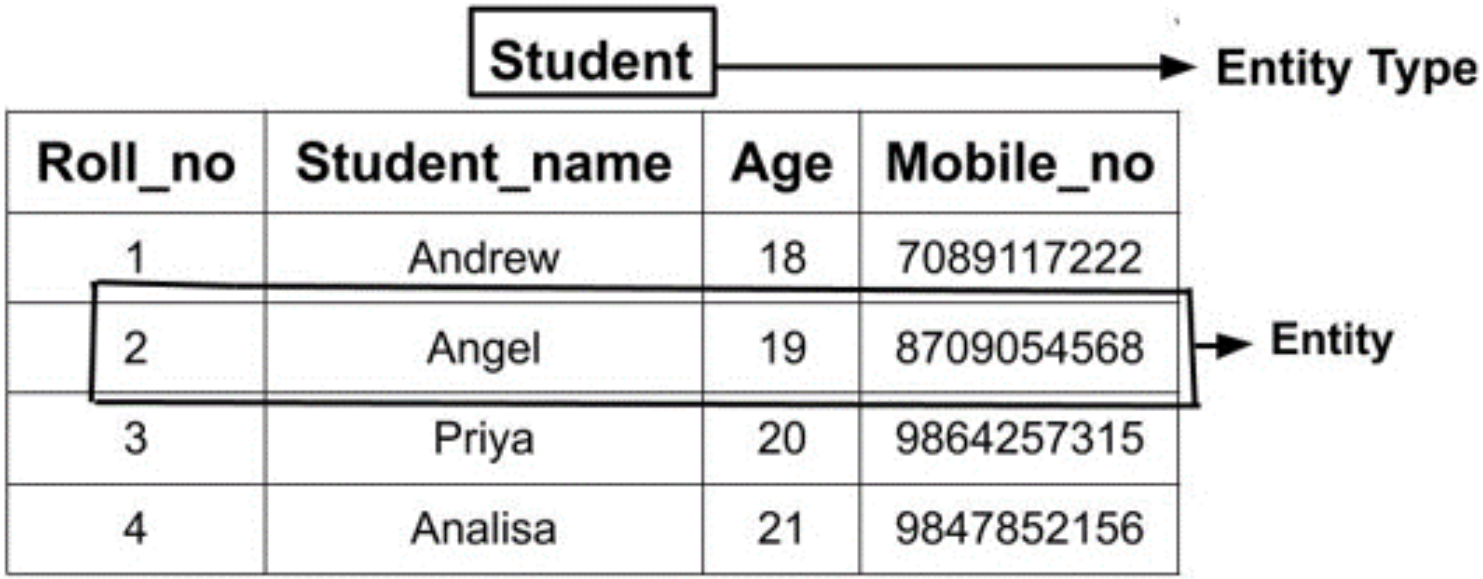


Figure 5: Entity Type

Types of Entity Type

CLARUSWAY
WAY TO REINVENT YOURSELF

Strong Entity Type: Strong entity are those entity types which has a key attribute. The primary key helps in identifying each entity uniquely.
Weak Entity Type:Weak entity type doesn't have a key attribute. Weak entity type can't be identified on its own. It depends upon some other strong entity for its distinct identity.

Example

There can be a room only if building exits.

Entity Set

Entity Set is a collection of entities of the same entity type.

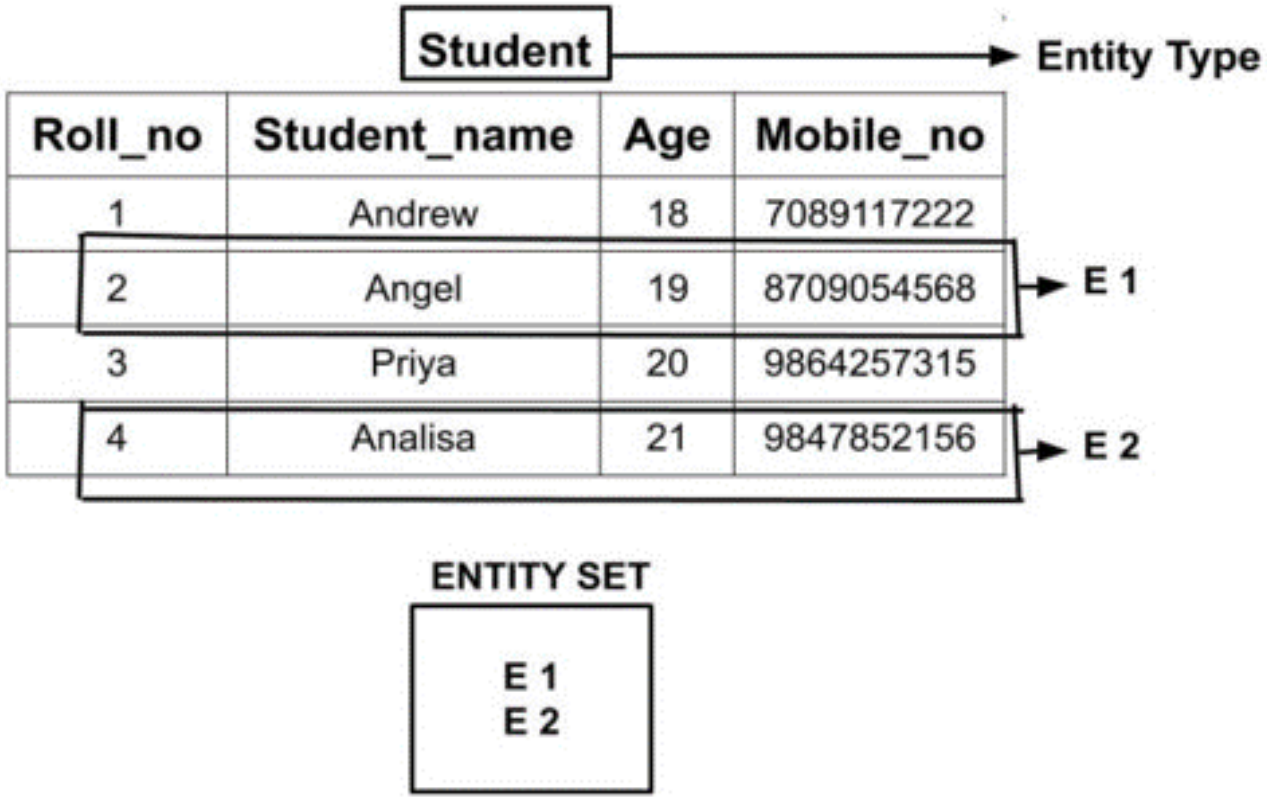


Figure 6: Entity Set

Attributes

Each entity is described by a set of attributes.

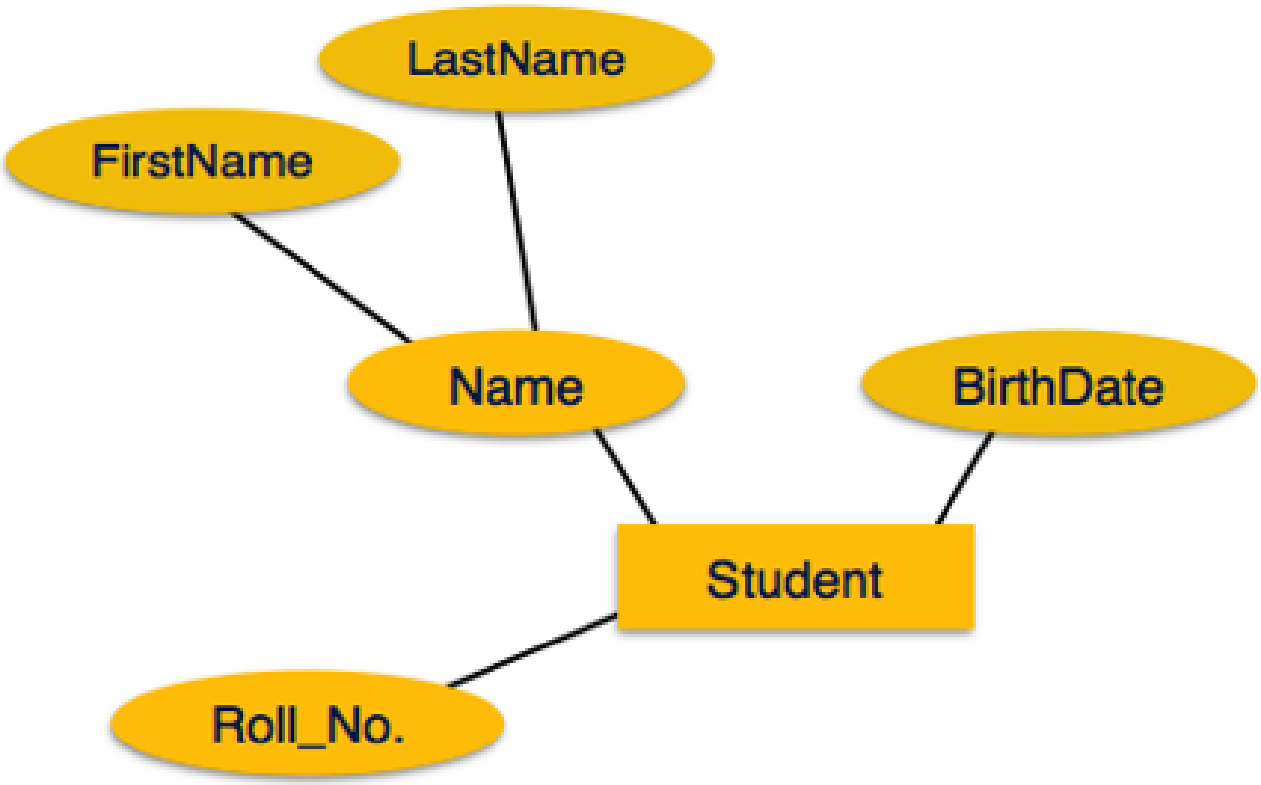


Figure 7: Attributes

Simple Attributes

CLARUSWAY
WAY TO REINVENT YOURSELF

A single-valued attribute is an attribute that can have only a single value. For example, a person can have only one Social Security number.

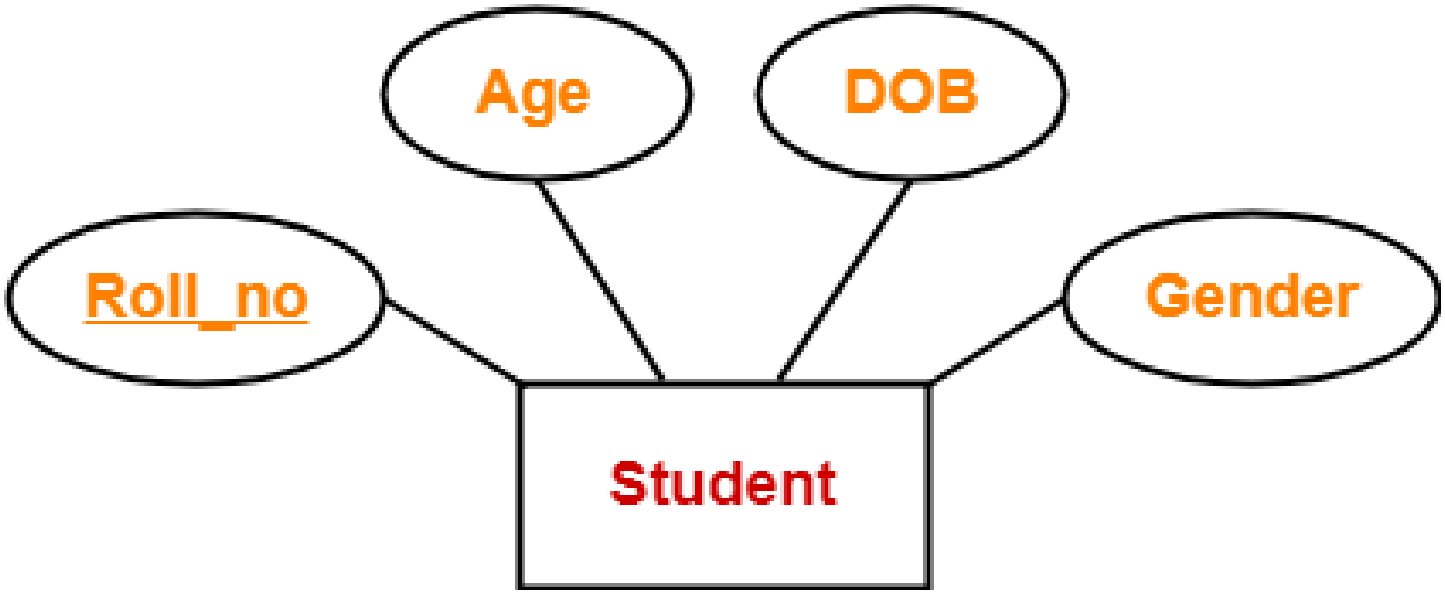


Figure 8: Simple Attributes

Composite Attributes

CLARUSWAY
WAY TO REINVENT YOURSELF

A composite attribute, not to be confused with a composite key, is an attribute that can be further subdivided to yield additional attributes. For example, the attribute ADDRESS can be subdivided into street, city, state, and zip code.

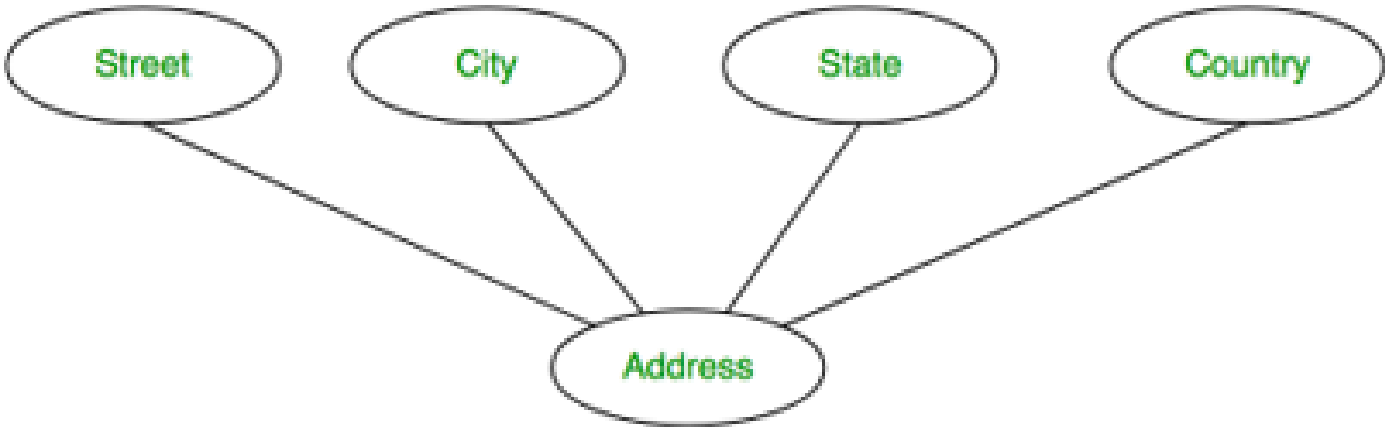


Figure 9: Composite Attributes

Multi-valued Attributes

CLARUSWAY
WAY TO REINVENT YOURSELF

Multivalued attributes are attributes that can have many values. For instance, a person may have several college degrees, and a household may have several different phones, each with its own number.

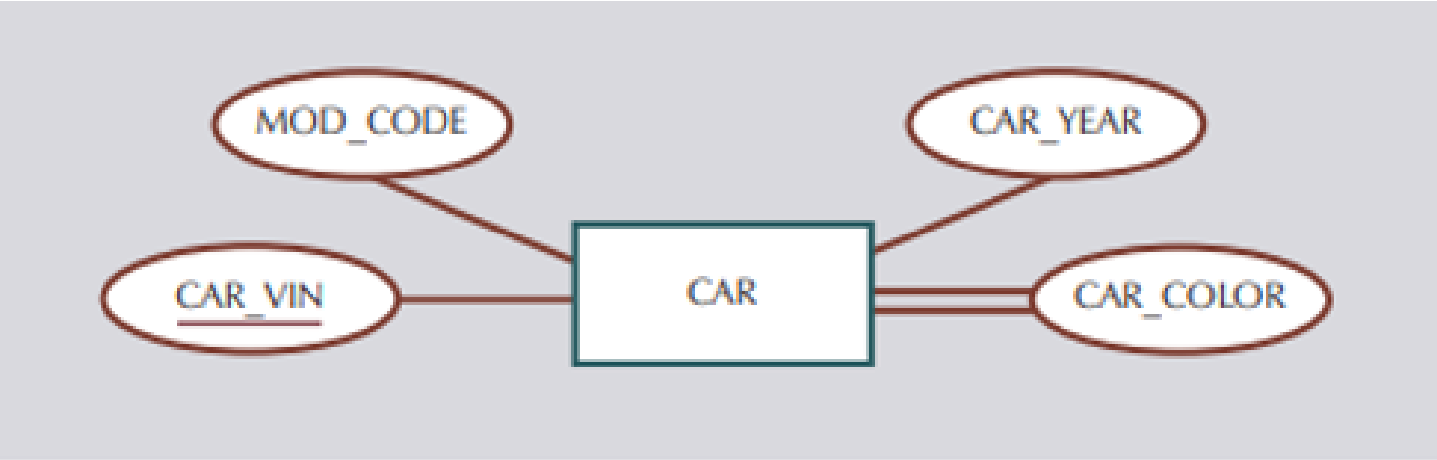


Figure 10: Multi-valued Attributes

Derived Attributes

A derived attribute is an attribute whose value is calculated (derived) from other attributes.

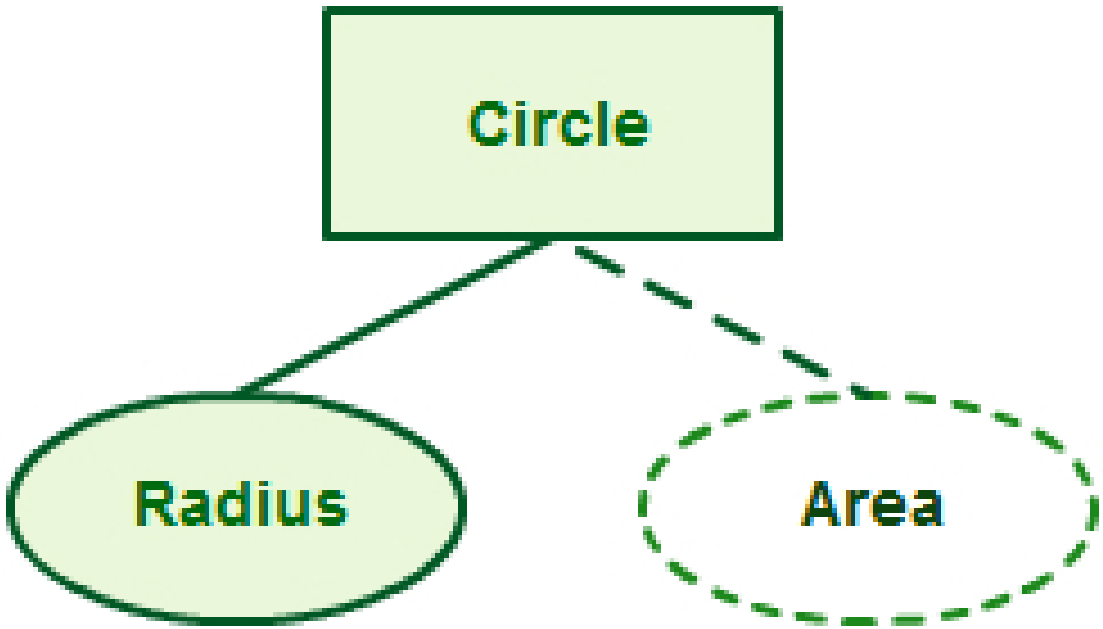


Figure 11: Derived Attributes

Null

A null is a special symbol, independent of data type, which means either unknown or inapplicable. It does not mean zero or blank.

- Can represent:
  - An unknown attribute value
  - A known, but missing, attribute value
  - A “not applicable” condition
- Can create problems when functions such as COUNT, AVERAGE and SUM are used.
- Can create logical problems when relational tables are linked.

Null

CLARUSWAY
WAY TO REINVENT YOURSELF

Student

Std_ID	First Name	Last Name	Email
1109	John	Price	jp@gmail.com
1401	Bran	NULL	bw@gmail.com
1756		Natan	cn@gmail.com
1945	Tom	Black	tb@gmail.com
1987	Nick	Norris	nn@gmail.com

This is NOT NULL

This is NULL

Figure 12: Null

Keys

CLARUSWAY
WAY TO REINVENT YOURSELF

The key is an attribute or a group of attributes whose values can be used to uniquely identify an individual entity in an entity set.

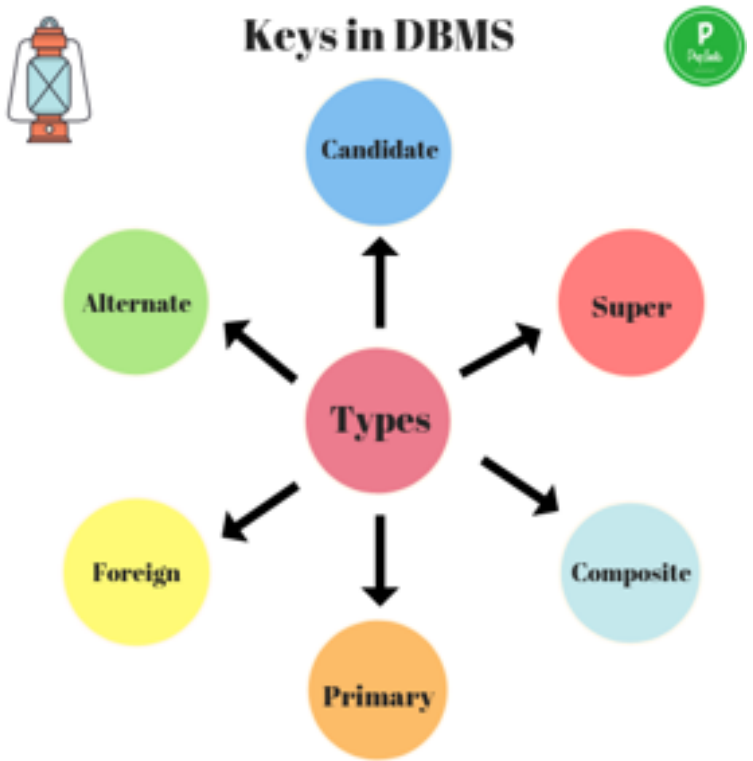


Figure 13: Keys

Keys

CLARUSWAY
WAY TO REINVENT YOURSELF

- A candidate key is a simple or composite key that is unique and minimal. It is unique because no two rows in a table may have the same value at any time. It is minimal because every column is necessary in order to attain uniqueness.
- A composite key is composed of two or more attributes, but it must be minimal.
- The primary key is a candidate key that is selected by the database designer to be used as an identifying mechanism for the whole entity set. It must uniquely identify tuples in a table and not be null.

Keys

CLARUSWAY
WAY TO REINVENT YOURSELF

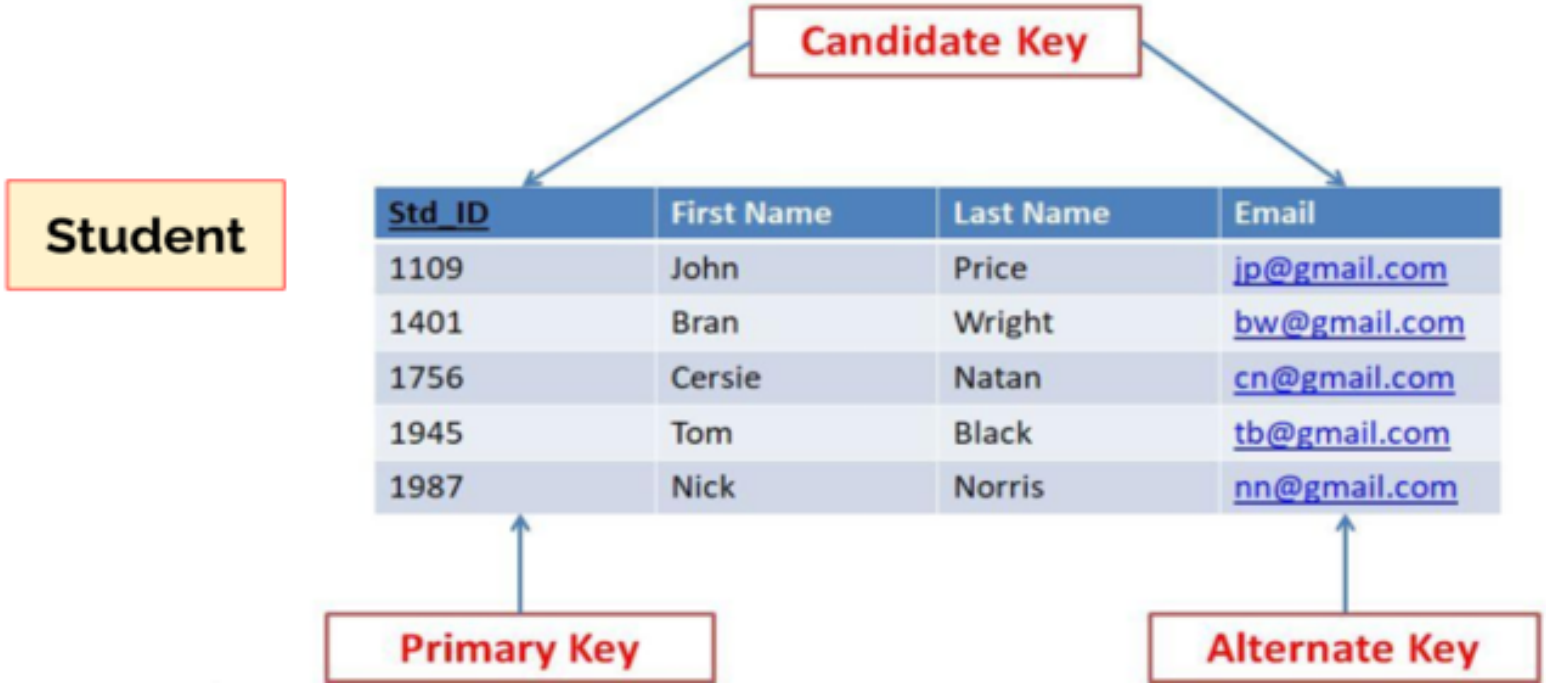


Figure 14: Candidate Key, Primary Key, Alternate Key

Keys

CLARUSWAY
WAY TO REINVENT YOURSELF

- A secondary key is an attribute used strictly for retrieval purposes (can be composite), for example: Phone and Last Name.
- Alternate keys are all candidate keys not chosen as the primary key.
- A foreign key (FK) is an attribute in a table that references the primary key in another table OR it can be null.

Keys

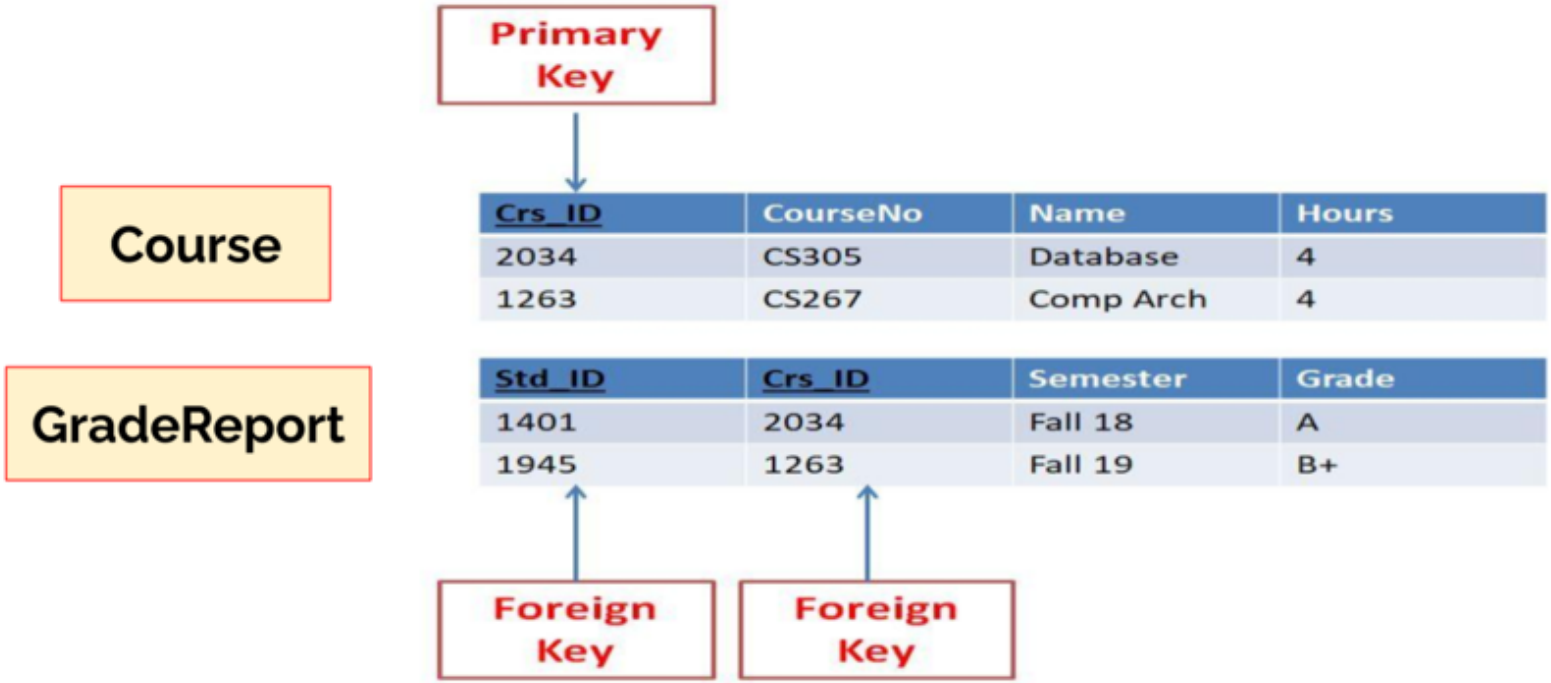


Figure 15: Primary Key, Foreign Key

Keys

CLARUSWAY
WAY TO REINVENT YOURSELF

STUDENT

STUD_NO	STUD_NAME	STUD_PHONE	STUD_STATE	STUD_COUNTRY	STUD_AGE
1	RAM	9716271721	Haryana	India	20
2	RAM	9898291281	Punjab	India	19
3	SUJIT	7898291981	Rajsthan	India	18
4	SURESH		Punjab	India	21

Table 1

STUDENT_COURSE

STUD_NO	COURSE_NO	COURSE_NAME
1	C1	DBMS
2	C2	Computer Networks
1	C2	Computer Networks

Table 2

Figure 16: Keys

Relationships

Relationship is an association between entities. The entities that participate in a relationship are also known as participants, and each relationship is identified by a name that describes the relationship. The relationship name is an active or passive verb.

Example

- a STUDENT takes a CLASS,
- a PROFESSOR teaches a CLASS,
- a DEPARTMENT employs a PROFESSOR,
- a DIVISION is managed by an EMPLOYEE and
- an AIRCRAFT is flown by a CREW.

Weak Relationship

CLARUSWAY
WAY TO REINVENT YOURSELF

A weak, or non-identifying, relationship exists if the primary key of the related entity does not contain a primary key component of the parent entity.

Course

Crs_ID	CourseNo	Name	Hours
2034	CS305	Database	4
1263	CS267	Comp Arch	4

Class

Class_ID	Crs_ID	Section	Time
10012	2034	1	Friday, 8:00-10:00 a.m.
10017	1263	3	Wednesday, 2:30-3:45 p.m.

Figure 17: Weak Relationship

Strong Relationship

CLARUSWAY
WAY TO REINVENT YOURSELF

A strong, or identifying, relationship exists when the primary key of the related entity contains the primary key component of the parent entity.

Course

Crs_ID	CourseNo	Name	Hours
2034	CS305	Database	4
1263	CS267	Comp Arch	4

Class

Crs_ID	Section	Time
10012	1	Friday, 8:00-10:00 a.m.
10017	3	Wednesday, 2:30-3:45 p.m.

Figure 18: Strong Relationship

One to One (1 : 1) Relationship

A one to one (1 : 1) relationship is the relationship of one entity to only one other entity, and vice versa. It should be rare in any relational database design.

Example

In nature, each element has one formula and each formula represents one element.

Figure 19: One to One (1 : 1) Relationship

One to Many $(1 : M)$ Relationship

CLARUSWAY
WAY TO REINVENT YOURSELF

A one to many $(1 : M)$ relationship should be the norm in any relational database design and is found in all relational database environments.

Example

In business, each staff works in one department and in each department, many staff can be working.

Figure 20: One to Many $(1 : M)$ Relationship

Many to Many $(M : N)$ Relationship

CLARUSWAY
WAY TO REINVENT YOURSELF

One entity from A can be associated with more than one entity from B and vice versa.

Example

In university, each student can attend many courses and in each course, there can be many students.

Figure 21: Many to Many $(M : N)$ Relationship

Breaking Up $M : N$ Relationship

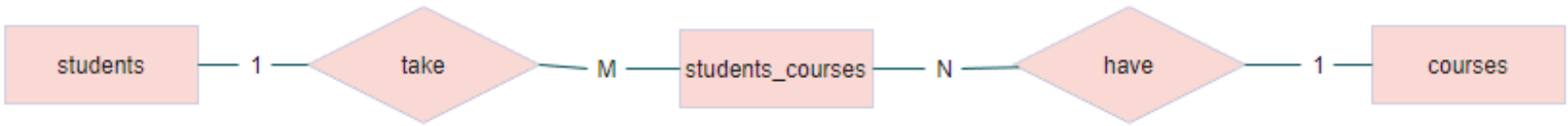


Figure 22: Breaking Up $M : N$ Relationship

Unary Relationship (Recursive)

A unary relationship, also called recursive, is one in which a relationship exists between occurrences of the same entity set. In this relationship, the primary and foreign keys are the same, but they represent two entities with different roles.

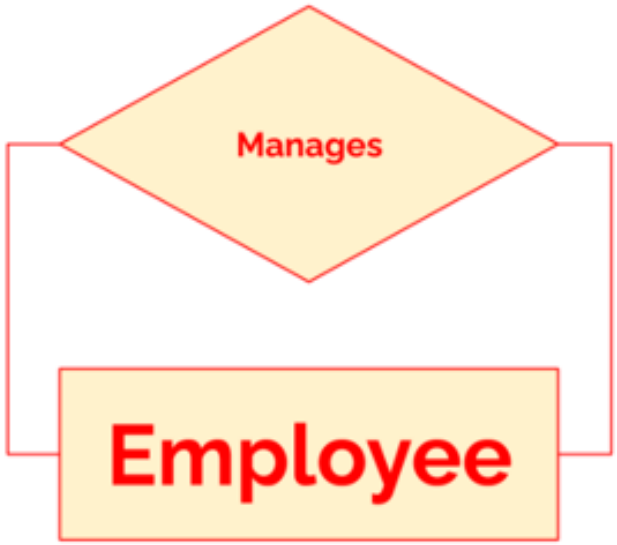


Figure 23: Unary Relationship (Recursive)

Unary Relationship (Recursive)

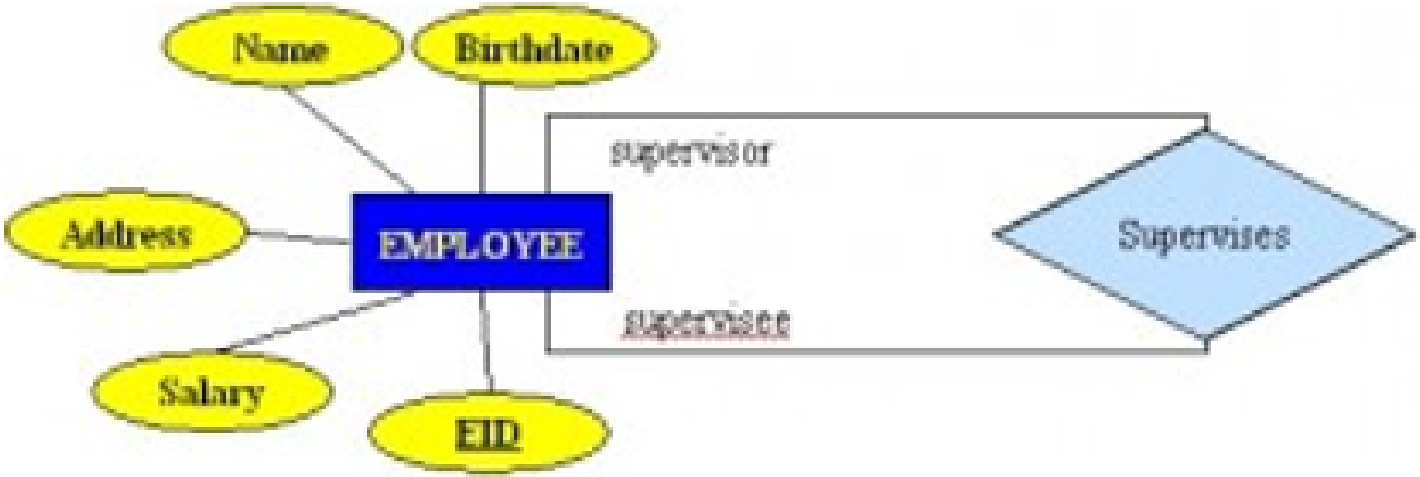


Figure 24: Unary Relationship (Recursive)

Ternary Relationship

CLARUSWAY
WAY TO REINVENT YOURSELF

A ternary relationship is a relationship type that involves many to many relationships between three tables.

Example

Let us assume there is some association among entities Technician, Project, and Notebook. If each technician can be working on any of several projects and using the same notebooks on each project, then three many-to-many binary relationships can be defined.

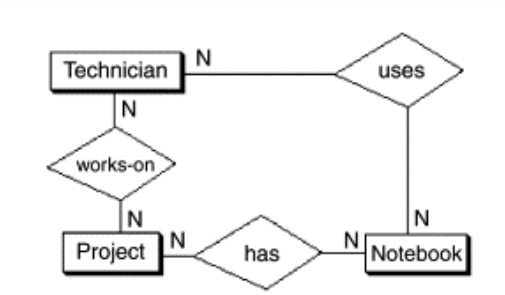


Figure 25: Ternary Relationship

Integrity Rules and Constraints

Domain Integrity

CLARUSWAY
WAY TO REINVENT YOURSELF

Domain restricts the values of attributes in the relation and is a constraint of the relational model. We need more specific ways to state what data values are or are not allowed and which format is suitable for an attribute. For example, the Employee ID (EID) must be unique or the employee Birthdate is in the range [Jan 1, 1950, Jan 1, 2000].

ID	NAME	SEMENSTER	AGE
1000	Tom	1st	17
1001	Johnson	2nd	24
1002	Leonardo	5th	21
1003	Kate	3rd	19
1004	Morgan	8th	A

Not allowed. Because AGE is an integer attribute

Figure 26: Domain Integrity

Entity Integrity

To ensure entity integrity, it is required that every table have a primary key. Neither the PK nor any part of it can contain null values. This is because null values for the primary key mean we cannot identify some rows.

EMPLOYEE

EMP_ID	EMP_NAME	SALARY
123	Jack	30000
142	Harry	60000
164	John	20000
	Jackson	27000

Not allowed as primary key can't contain a NULL value

Figure 27: Entity Integrity

Referential Integrity

CLARUSWAY
WAY TO REINVENT YOURSELF

Referential integrity requires that a foreign key must have a matching primary key or it must be null. This constraint is specified between two tables (parent and child); it maintains the correspondence between rows in these tables.

Crs_ID	CourseNo	Name	Hours
2034	CS305	Database	4
1263	CS267	Comp Arch	4

Std_ID	Crs_ID	Semester	Grade
1401	2034	Fall 18	A
1945	1263	Fall 19	B+
2654	2015	Fall 19	B

This value is not allowed because this value is not defined as a primary key in the course table.

Figure 28: Referential Integrity

Foreign Key Rules

Additional foreign key rules may be added when setting referential integrity, such as what to do with the child rows (in the Orders table) when the record with the PK, part of the parent (Customer), is deleted or changed (updated).

Example

- Update Cascade
- Delete Cascade

Enterprise Constraints

CLARUSWAY
WAY TO REINVENT YOURSELF

Enterprise constraints – sometimes referred to as semantic constraints – are additional rules specified by users or database administrators and can be based on multiple tables.

Example

- A class can have a maximum of 30 students.
- A teacher can teach a maximum of four classes per semester.

Business Rules

Business rules are obtained from users when gathering requirements. The requirements-gathering process is very important, and its results should be verified by the user before the database design is built.

Example

- A teacher can teach many students.
- A class can have a maximum of 35 students.
- A course can be taught many times, but by only one instructor.

Cardinality and Connectivity

- Business rules are used to determine cardinality and connectivity. Cardinality describes the relationship between two data tables by expressing the minimum and maximum number of entity occurrences associated with one occurrence of a related entity.
- The outermost symbol of the relationship symbol, on the other hand, represents the connectivity between the two tables. Connectivity is the relationship between two tables, e.g., one to one or one to many. The only time it is zero is when the FK can be null.

Cardinality and Connectivity

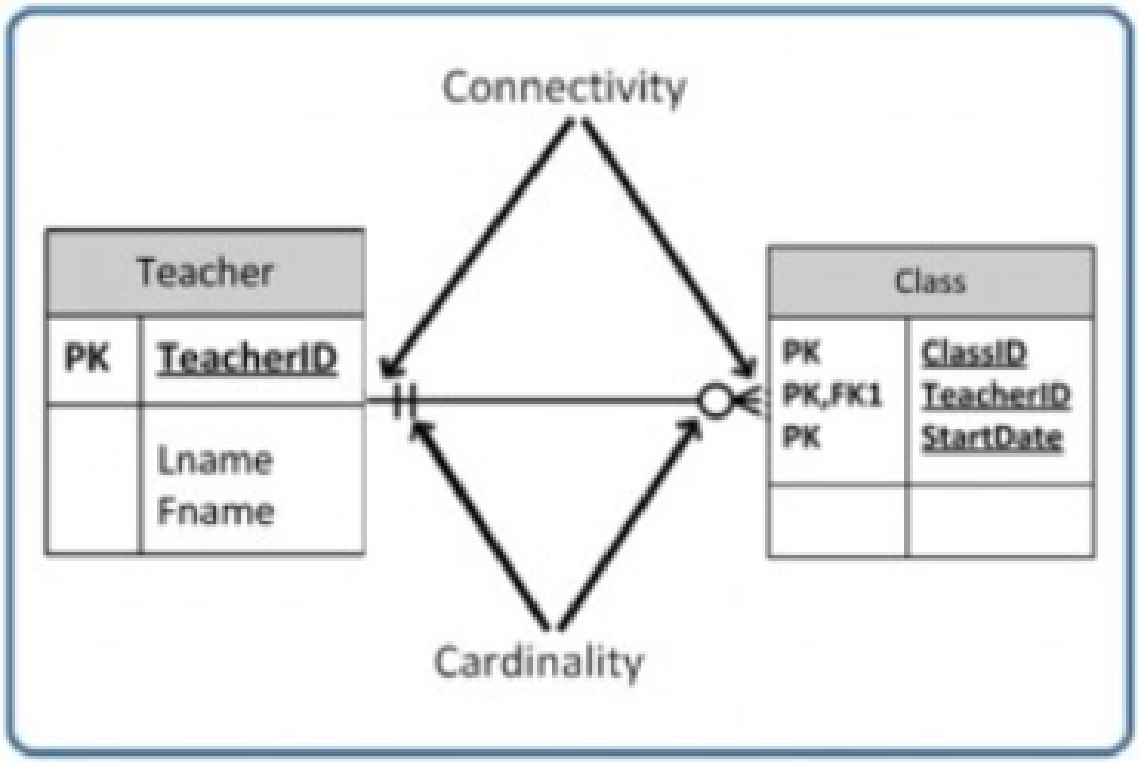
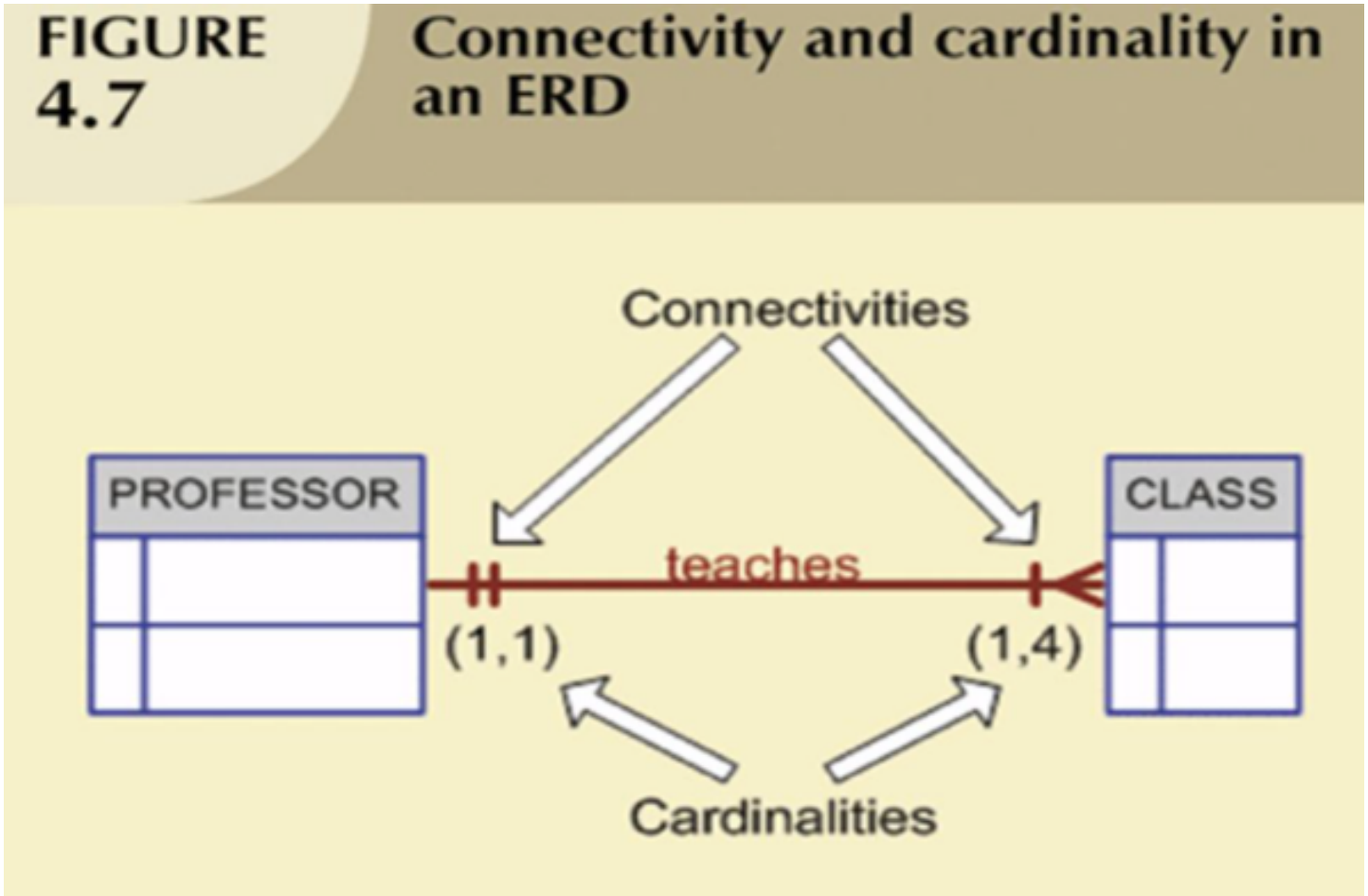


Figure 29: Cardinality and Connectivity

Cardinality and Connectivity



Types of Relationship

CLARUSWAY
WAY TO REINVENT YOURSELF

- The line that connects two tables, in an ERD, indicates the relationship type between the tables: either identifying or non-identifying. An identifying relationship will have a solid line (where the PK contains the FK).
- A non-identifying relationship is indicated by a broken line and does not contain the FK in the PK.

Types of Relationship

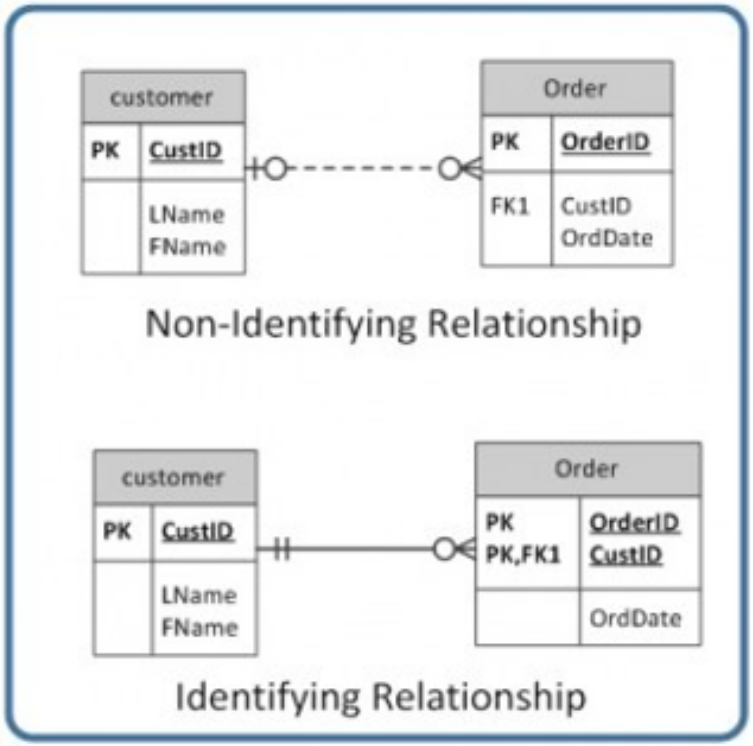


Figure 30: Identifying and Non-Identifying Relationship

Types of Relationship

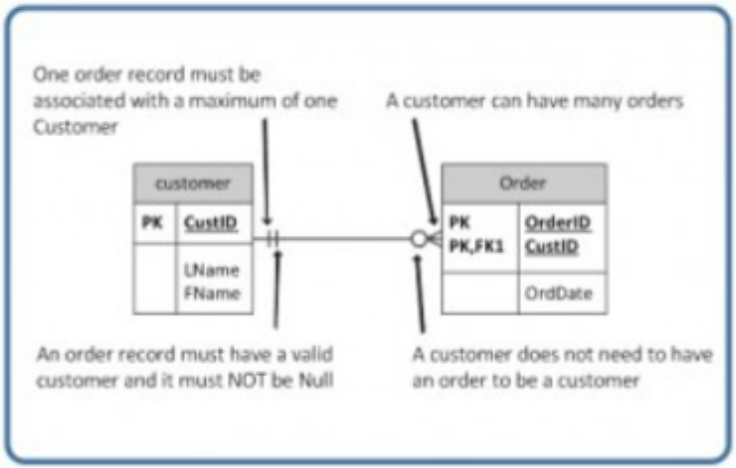
CLARUSWAY
WAY TO REINVENT YOURSELF

Optional relationships: In an optional relationship, the FK can be null or the parent table does not need to have a corresponding child table occurrence.
Mandatory relationships: In a mandatory relationship, one entity occurrence requires a corresponding entity occurrence. The symbol for this relationship shows one and only one.

Types of Relationship

CLARUSWAY
WAY TO REINVENT YOURSELF

Optional Relationship:



Mandatory Relationship:

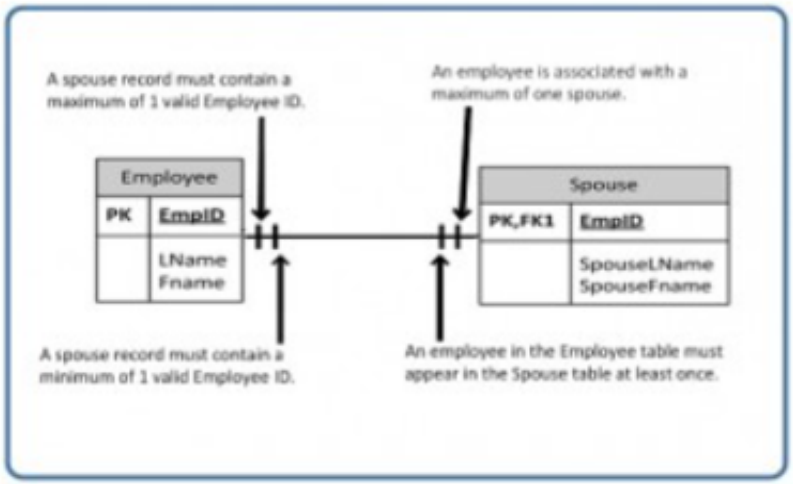
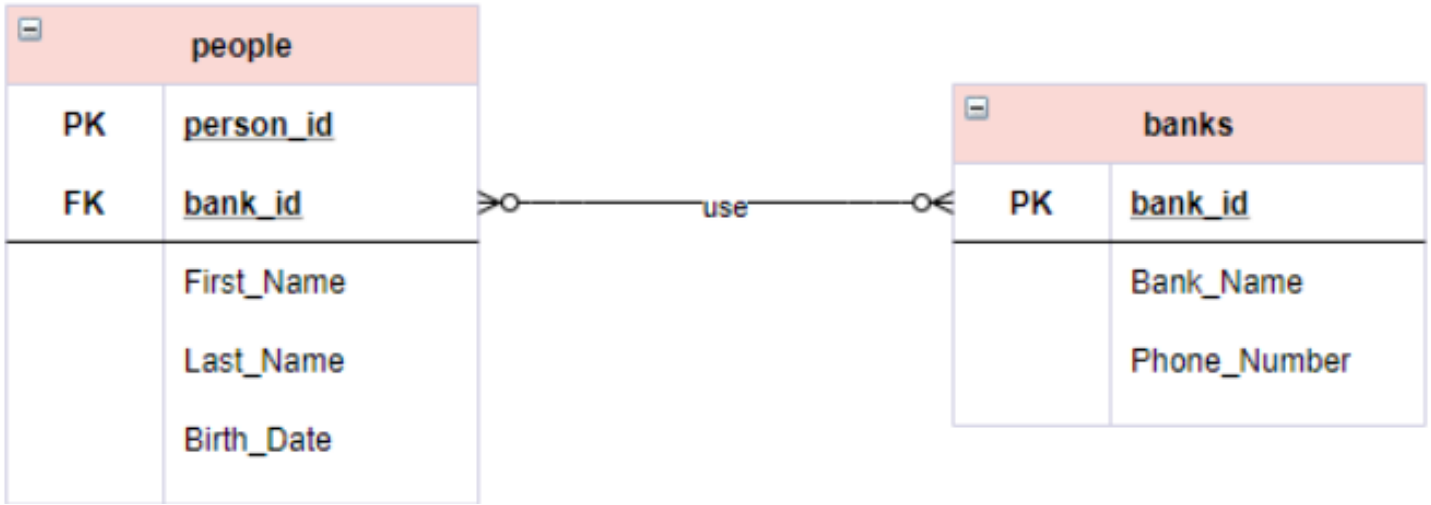
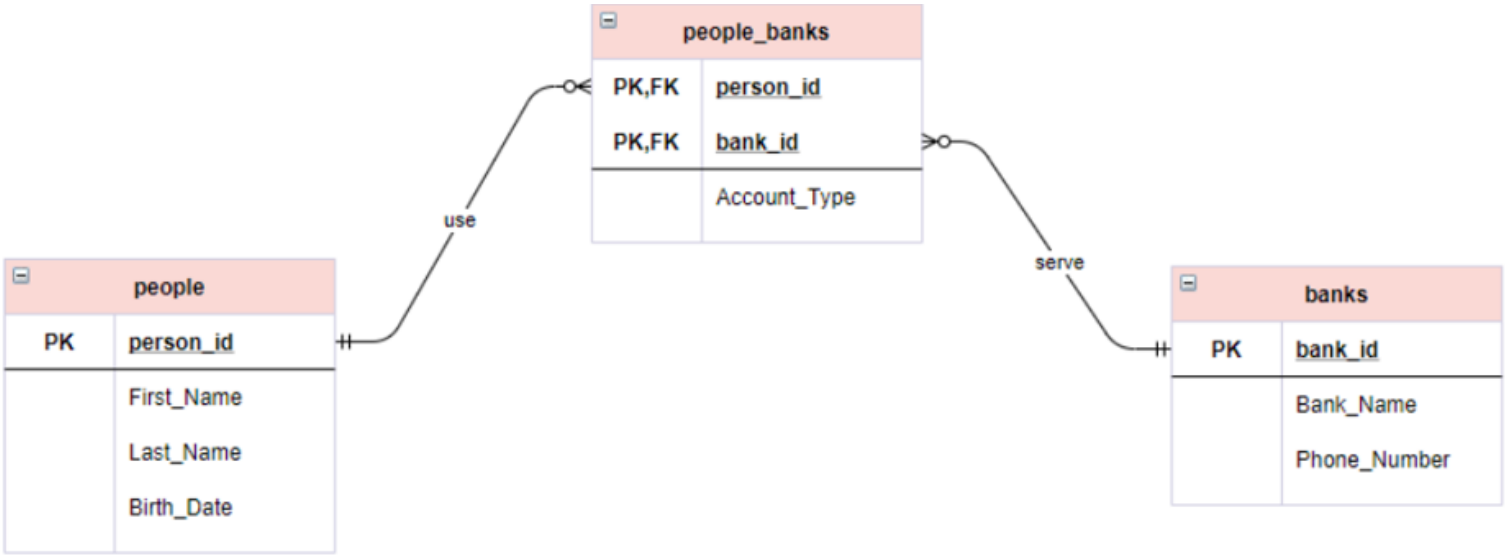


Figure 31: Optional and Mandatory Relationship

Many to Many Examples



Many to Many Examples



Many to Many Examples

CLARUSWAY
WAY TO REINVENT YOURSELF

people

person_id	bank_id	First_Name	Last_Name	Birth_Date
1	1	Jack	London	20.05.1989
1	2	Jack	London	20.05.1989
2	3	Harry	Larry	12.07.1974
3	1	Henry	Jackson	13.02.1990
2	2	Harry	Larry	12.07.1974

banks

bank_id	Bank_Name	Phone_Number
1	X bank	3330777
2	Y bank	5551444
3	Z bank	6662888

Many to Many Examples

CLARUSWAY
WAY TO REINVENT YOURSELF

people

person_id	First_Name	Last_Name	Birth_Date
1	Jack	London	20.05.1989
2	Paul	Kane	30.04.1965
3	Harry	Larry	12.07.1974
4	Henry	Jackson	13.02.1990
5	Mary	Dune	16.01.1980

people_banks

person_id	bank_id	Account_Type
1	1	Checking Account
1	2	Savings Account
2	1	Individual Retirement Account
2	3	Checking Account
3	3	Checking Account

banks

bank_id	Bank_Name	Phone_Number
1	X bank	3330777
2	Y bank	5551444
3	Z bank	6662888